POLITECNICO DI TORINO

Corso di Laurea in Ingegneria del Cinema e dei Mezzi di Comunicazione

Tesi di Laurea Magistrale

CuBee: progettazione e sviluppo di un sistema interattivo modulare

Relatore
prof. Giovanni Malnati

Laureandi
Elisa Musolino
Stelian Spulber

Anno accademico 2018 – 2019

Indice

Introduzione **7**

I Dimensione virtuale **10**

1 L'evoluzione delle tecniche di animazione **13**
1.1 L'animazione digitale 2D 23
1.2 La versatilità dell'animazione digitale 26

2 Le fasi di lavoro: Pre produzione **37**
2.1 L'idea del cortometraggio 37
2.2 Soggetto e sceneggiatura 41
2.2.1 Il viaggio dell'eroe 49
2.3 Character design e sfondi 54
2.4 Storyboard e animatic 63

3 Le fasi di lavoro: Produzione **67**
3.1 Moho 68
3.2 Disegno e colorazione 71
3.3 Rigging 82
3.4 Animazione 85
3.4.1 I principi dell'animazione 86
3.4.2 Walk cycle 95
3.4.3 Il volo del gabbiano 97

II Dimensione fisica **101**

4 Il cubo: storie di interfacce tangibili **103**
4.1 Sul cubo 103
4.2 Case studies 106
4.2.1 Sifteo 106

4.2.2 Radica Cube World 110

5 Progettazione: dall'idea al prototipo fisico 113
5.1 Requisiti 114
5.2 Prototipazione 116
5.3 CAD 3D 118
5.3.1 Autodesk Fusion 360 122
5.4 Stampa 3D 129
5.4.1 Anycubic i3 Mega S 137
5.5 L'evoluzione del dispositivo 140

6 Progettazione hardware 145
6.1 Elaborazione: dai microcontrollori ai microcomputer 145
6.1.1 Raspberry Pi Zero W 149
6.2 Comunicazione seriale 153
6.2.1 SPI 154
6.2.2 I^2C 158
6.3 Grafica 160
6.3.1 Display LCD 2.8" ILI9341 164
6.3.2 LED SMD5050 WS2812 166
6.4 Interazione: Touch 169
6.4.1 Modulo Texas Instruments ADS7846 172
6.5 Interazione: Motion sensing 176
6.5.1 IMU Adafruit LSM9DS1 177
6.5.2 Tilt sensing e shake 182
6.6 Interazione: Modularità 187
6.6.1 Lettore NFC Elechouse PN532 v3 190
6.7 Suono 194
6.7.1 I^2S 195
6.7.2 Amplificatore Adafruit MAX98357A 196
6.7.3 Speaker CUI CDS-20144-L100 198
6.8 Alimentazione 199
6.8.1 Boost converter Adafruit PowerBoost 1000C 199

7 Progettazione software 203
7.1 Comunicazione tra i cubi 204
7.2 Implementazione della modularità 211
7.3 Motion sensing 214
7.4 Illuminazione delle facce laterali 220
7.5 Sviluppo di videogiochi con Pygame 224

III Dimensione interattiva **229**

8 Progettazione dell'interazione **231**
8.1 Il contesto museale . . . 231
8.2 Storytelling interattivo . . . 239
8.3 Design dell'esperienza . . . 244
8.3.1 Struttura della narrazione . . . 245
8.4 Videogiochi . . . 252
8.4.1 Unlock the cube . . . 257
8.4.2 Ocean Getaway . . . 259
8.4.3 Save Puppy . . . 262
8.4.4 End the Strike . . . 266
8.5 Implementazione software . . . 271

9 Conclusioni **277**
9.1 Sviluppi futuri . . . 281

Bibliografia **283**

Introduzione

A partire dal movimento ecologista radicale Extinction Rebellion e soprattutto dallo slogan *Skolstrejk för klimatet*, «Sciopero della scuola per il clima», lanciato nel 2018 dall'allora quindicenne Greta Thunberg, si è sviluppata, a livello globale, una conversazione intorno agli effetti dei cambiamenti climatici. Attraverso questa tesi si è voluto inserirsi in tale conversazione, con lo scopo di contribuire a sensibilizzare più persone possibili, a partire dalle nuove generazioni. Questo intento si è concretizzato nella progettazione del sistema CuBee, pensato per un'installazione interattiva che verrà realizzata in uno spazio museale.

L'obiettivo della tesi è stato quello di progettare, realizzare e valutare l'efficacia di un tipo di comunicazione appartenente alla sfera dell'*edutainment* e veicolata attraverso contenuti audiovisivi come un cortometraggio di animazione ed un'interfaccia tangibile che permette una partecipazione attiva dell'utente all'interno di un contesto definibile come *interactive storytelling*.

CuBee è un progetto multidisciplinare e risponde ai seguenti tre aspetti che abbiamo voluto sperimentare:

1. progettare e sviluppare un dispositivo di gioco, comprensivo degli aspetti legati al design, alle dinamiche di gioco e della creazione dei contenuti;
2. cimentarci nella produzione completa, dall'idea alla postproduzione, di un video animato;
3. progettare un'interazione tra il *virtuale* e il *tangibile* unendo entrambi questi aspetti in un'installazione interattiva organizzata in un contesto museale, come quello del Museo A come Ambiente (MAcA) di Torino.

CuBee nasce quindi come progetto multidisciplinare basato su un sistema che prevede un oggetto manipolabile nello spazio fisico, la cui interazione può modificare lo stato e il comportamento di uno spazio virtuale. Nello specifico il progetto si articola in tre parti, ognuna delle quali pone l'attenzione su una specifica dimensione: virtuale, fisica e interattiva.

La prima parte della tesi consiste in un approfondimento sullo stato dell'arte delle tecniche di animazione. In seguito si discutono le varie fasi della realizzazione del cortometraggio, attraverso il brainstorming per le idee, la fase di ricerca e documentazione sull'argomento trattato, la creazione dei disegni, lo studio dei personaggi della storia e l'animazione.

Nella seconda parte è presente una breve analisi di alcuni dispositivi di gioco assimilabili al nostro concetto, proseguendo con la descrizione delle fasi di realizzazione dell'oggetto fisico. Nello specifico vengono trattate le caratteristiche dei componenti elettronici utilizzati e le scelte effettuate in ambito hardware e software nonché la modellazione 3D del prototipo e il processo di stampa di quest'ultimo.

La terza parte è incentrata sull'integrazione delle due fasi precedenti e quindi sul meccanismo narrativo di storytelling interattivo scelto per creare l'intero sistema CuBee. Sono inoltre illustrati i contenuti e il funzionamento del sistema con riguardo alle grafiche esplicative e ai software utilizzati per l'implementazione complessiva.

La tesi si conclude con la descrizione della progettazione dell'installazione interattiva.

Parte I

Dimensione virtuale

Capitolo 1

L'evoluzione delle tecniche di animazione

— Elisa Musolino

L'atto di osservare e la propensione a replicare ciò che viene visualizzato nella quotidianità è una caratteristica innata dell'uomo. In particolare, tale inclinazione a raffigurare concetti o a crearne di nuovi, è maggiormente sviluppata nella figura dell'artista. Una figura che, spinta dalla creatività e dalla volontà di esprimersi, fa dell'animazione un potente mezzo di comunicazione attraverso cui trasmettere emozioni capaci di raggiungere qualsiasi persona superando le barriere linguistiche.

Dal punto di vista semantico, il concetto di animazione, animare dal latino 'dare vita', si può definire come l'illusione del movimento, creato a partire dalla visualizzazione di una serie di immagini disposte sequenzialmente. Ciascun frame della sequenza è un disegno o una fotografia che differisce di poco dal fotogramma precedente proprio per suscitare la sensazione di movimento nello spazio e nel tempo, una percezione definita come *effetto phi*. Tale nome è stato attribuito agli inizi del XX secolo da Max Wertheimer, psicologo della *Gestalt*, per descrivere una possibile interpretazione della percezione del movimento. Secondo la psicologia della Gestalt o psicologia della forma, "il tutto è più della somma delle singole parti". Si tratta di una teoria che infatti considera la percezione totale del movimento di singole parti, immagini, come una capacità innata di ciascun individuo, che a livello cerebrale raccoglie ed elabora gli stimoli visivi distanziati tra di loro, restituendo come risultato una visione continua ed uniforme.

> "Il grande problema dell'industria dei cartoni animati è stato quello di rendere il concatenamento dolce come quello del cinema 'fotografico', con un intento fondamentalmente realista. Perciò, sono state create diverse tecniche."[1]

Il primo tentativo di animazione risale già al periodo del Paleolitico, in cui venivano raffigurati sulla pietra, ad esempio animali con gambe sovrapposte per simulare la percezione del movimento. In altri casi i disegni apparivano come una effettiva sequenza di fotogrammi, come emerge dagli studi dell'archeologo Marc Azéma e l'artista Florent Rivère[2].
Un ulteriore esempio di simulazione di un'animazione si può ritrovare anche nell'antico Egitto, come il murale risalente a 4000 anni fa, che raffigura un incontro di lotta attraverso una sequenza di immagini, visibile in figura 1.3.

Figura 1.1: M. Azema, J. Clottes, Chauvet Cave Scientific Team. Fonte: Il Fatto Storico.

Figura 1.2: Gruppo di cavalli disegnato nella grotta Chauvet, Francia. Fonte: Wikipedia.

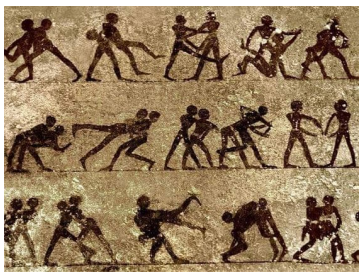


Figura 1.3: Tomba di Khnumhotep, cimitero di Beni Hassan, Egitto. Fonte: Pinterest/ Art Of Combat.

Diversi furono gli strumenti creati per simulare il movimento di immagini fisse, come l'invenzione della *Lanterna Magica*, del *Fenachistoscopio*, dello *Zootropio* e del *Prassinoscopio*.

L'ideazione della lanterna magica del 1659, visibile in figura 1.4, ha origini incerte. Secondo quanto riportato nel libro *Mondi possibili. Un viaggio nella storia del cinema d'animazione*[3], lo scrittore, regista e specialista nella preistoria del cinema, Laurent Mannoni, ha dimostrato che l'ideazione di tale dispositivo non è da attribuire al padre gesuita Athanasius Kircher, come si ritiene, ma si tratta di un'invenzione appartenente allo scienziato olandese Christiaan Huygen. Egli, ispirandosi alle figure della *Danza della morte* di Holbein, dipinse due lastre di vetro: su quella fissa, riprodusse uno scheletro senza cranio e braccio destro, mentre sulla lastra mobile disegnò solo il cranio e il braccio. I due vetri sovrapposti erano disposti all'interno di una piccola scatola dalle pareti oscurate, ossia la lanterna. All'interno di questa era presente una candela o una lampada ad olio utilizzata come fonte di luce. I raggi luminosi, colpendo uno specchio concavo posteriore, erano riflessi prima su di un condensatore e poi sulla lastra. In questo modo, davanti all'obiettivo le immagini erano proiettate ingrandite e muovendo la lastra mobile verso l'alto e verso il basso, si aveva l'impressione che lo scheletro muovesse il cranio

e il braccio. La reazione incredula degli spettatori inconsapevoli alla vista di queste immagini proiettate portò a denominare il dispositivo come lanterna magica.

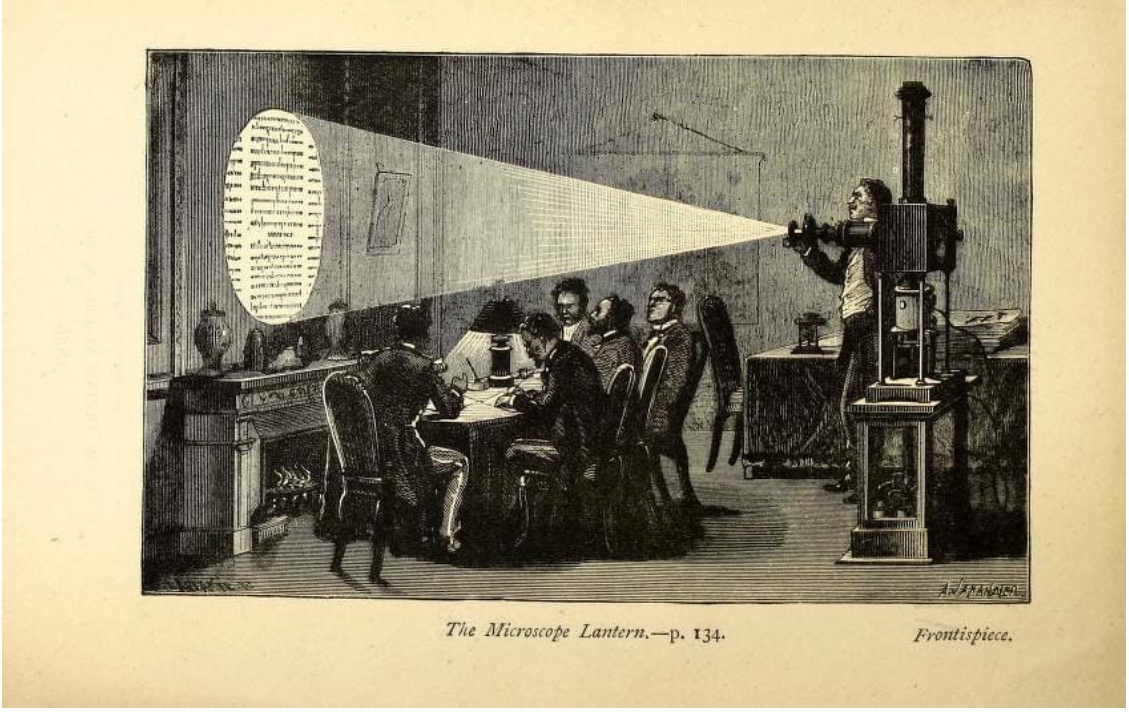


Figura 1.4: Illustrazione della lanterna magica di Christiaan Huygens. Fonte: The magic lantern manual, W.I.Chadwick, Internet archive.

Tale tecnica introdusse così il dinamismo e la proiezione ingrandita che costituiscono due degli elementi fondamentali del cinema di animazione.
Huygens però, non diede molta importanza a tale sperimentazione; fu Kircher che invece ne comprese l'enorme potenziale tanto da utilizzare la lanterna magica come strumento di intrattenimento nel cosiddetto cinematografo.

Il Fenachistoscopio, in figura 1.5, inventato da Joseph Plateau nel 1832, è considerato il primo vero dispositivo di animazione in grado di creare un'illusione fluida del movimento. Tale strumento consiste in due dischi rotanti attaccati verticalmente a una maniglia. Uno di questi ha delle fessure radiali equidistanti attraverso le quali l'osservatore può guardare il secondo disco che contiene una sequenza di immagini.

Lo Zootropio, in figura 1.6, fu invece inventato nel 1834 da William George Horner. Si tratta di uno strumento ottico costituito da un cilindro rotante intorno al suo asse di simmetria, con feritoie poste a intervalli regolari, una per ogni immagine, attraverso cui vedere la rapida successione delle immagini disegnate che creano l'illusione del movimento. Una simile struttura rende però le immagini deformate rispetto alla realtà.

Questo effetto visivo è usato ancora oggi nella realizzazione di GIF animate.

Figura 1.5: Disco di Fenachistoscopio.
Fonte: Rai News.

Figura 1.6: Zootropio vittoriano.
Fonte: Intelligent Heritage.

Nel 1877, il francese Charles Émile Reynaud inventò il Prassinoscopio, visibile in figura 1.7. Tale dispositivo è costituito da una striscia di immagini disposte in modo circolare sulla superficie di un cilindro girevole collegato a un perno e permette la proiezione in successione delle immagini.

Si tratta di una versione migliorata dello Zootropio poiché sostituisce le feritoie con una serie di specchi posizionati lungo il cilindro interno e orientati verticalmente di 45° in modo da riflettere le immagini verso l'osservatore, fornendo immagini più chiare.

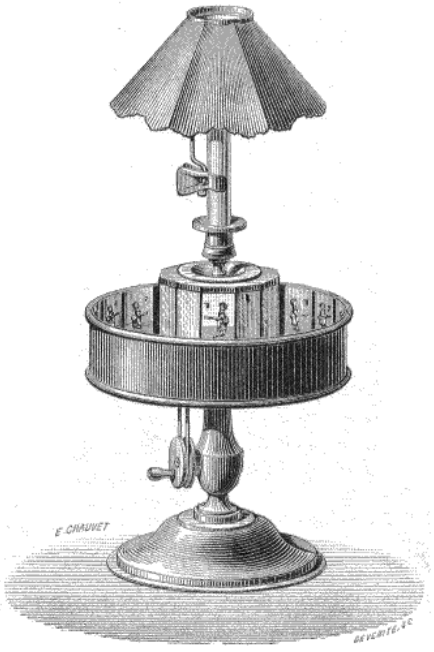


Figura 1.7: Illustrazione del Prassinoscopio di Reynaud.
Fonte: Wikipedia.

Nel 1888, lo stesso Charles Émile Reynaud, dopo aver perfezionato il *prassinoscopio da proiezione*, un'evoluzione del precedente strumento, diede vita al cosiddetto *Teatro ottico*, considerato il primo caso di cinema d'animazione. Tramite questo dispositivo, nel 1892 a Parigi, Reynad riuscì ad effettuare la prima proiezione in pubblico di immagini animate, dal nome *Pantomimes Lumineuses*[4]. Su delle piccole lastre di vetro erano dipinte a mano

delle figure e tali lastre erano collegate in sequenza su una banda posizionata su di un tamburo. Durante la rotazione di tale tamburo, veniva fatto passare un fascio di luce proveniente da una lanterna magica e tale fascio, attraversando le lastre era riflesso su degli specchi; infine, tramite una lente, le immagini erano proiettate su di uno schermo.

> "L'apparecchio ha come scopo quello di ottenere l'illusione del movimento, non più limitata alla ripetizione delle stesse pose a ogni giro dello strumento, come avviene necessariamente in tutti gli apparecchi conosciuti (zootropi, prassinoscopi ecc.) ma, al contrario, di varietà e durata indefinite e capace così di produrre vere scene animate con uno sviluppo illimitato. Da cui il nome di Teatro ottico dato dall'inventore a questo apparecchio."
>
> – *Charles Émile Reynaud*

Tre anni dopo, precisamente il 28 dicembre 1895 al Salon Indien del Café des Capucines a Parigi, nacque il cinematografo per merito di Auguste e Louis Lumière. Mediante la macchina da presa fu possibile sostituire il disegno manuale su ciascuna banda di celluloide del metodo di Reynaud con la tecnica *one turn, one picture*, ossia "un giro di manovella, un fotogramma" e che può essere considerata come la principale tecnica cartoon del periodo antecedente la nascita della computer grafica.

> "Nel caso dei disegni animati, la macchina da presa riprende un disegno per volta e da qui si parla di riprese 'a passo uno'; per un secondo di proiezione ne occorrono sedici, quindi quattromilaottocento disegni per un filmato di cinque minuti."[4]

Nel 1908, Émile Cohl realizzò quello che è considerato da molti come il primo cartone animato della storia, *Fantasmagorie*, in figura 1.8.

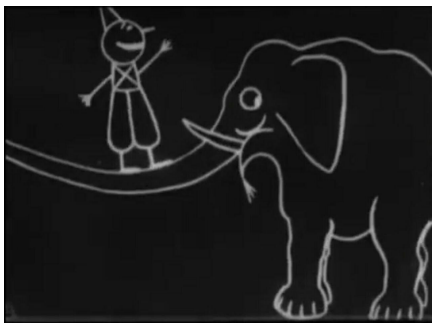


Figura 1.8: Fotogramma tratto da *Fantasmagorie*. Fonte: YouTube.

Si tratta di una composizione di 700 disegni su lastre di vetro, realizzati e fotografati singolarmente da Cohl stesso, in cui la rapida successione di queste lastre creava l'illusione del movimento di personaggi mutevoli. Tale lavoro richiedeva una notevole quantità di tempo, in quanto ciascuna illustrazione doveva essere disegnata da zero.

Ciò che permise di facilitare tale operazione e consentì di ottenere un movimento più fluido fu l'introduzione dei fogli di rodovetro nel 1914, ossia fogli di celluloide trasparenti che permettevano di avere gli sfondi inalterati rispetto ai personaggi. Questi ultimi infatti erano disegnati ognuno con posizioni diverse e contenenti soltanto le parti che si intendevano animare, ciascuno su fogli singoli da sovrapporre sugli sfondi di base.

Tale tecnica che fa uso di fogli di acetato, è comunemente conosciuta come animazione tradizionale o *cell animation*.

Tra i più importanti esempi di cartoni animati realizzati in modo tradizionale vi sono quelli realizzati dai colossali Walt Disney Studios, nati a Los Angeles nel 1923, che cambiarono in modo significativo il mondo dell'animazione. In pochi anni, Disney diede vita al primo cartone animato sonoro *Steamboat Willie* del 1928, seguito cinque anni dopo dal primo cartone a colori dal titolo *Flower and Trees*, ed infine il celebre lungometraggio *Biancaneve e i sette nani* del 1937. Tra i creatori di quest'ultimo film di animazione vi è Bill Tyla, che si è occupato dell'animazione del personaggio Brontolo ed è considerato uno dei migliori animatori della Disney nell'epoca d'oro dell'animazione di Hollywood, il primo ad aver portato emozioni vere sullo schermo dei cartoni animati. A tal proposito, è interessante riportare una citazione del libro *The illusion of life. Disney Animation*[5] in cui Tytla interpreta il significato dell'animazione:

> "The problem is not a single track one. Animation is not just timing, or just a well-drawn character, it is the sum of all the factors named. No matter what the devil one talks about- whether force or form, or well-drawn characters, timing, or spacing- animation is all these things- not any one. What you as an animator are interested in is conveying a certain feeling you happen to have at that particular time. You do all sorts of things in order to get it. Whether you have to rub out a thousand times in order to get it is immaterial."

Il metodo dell'animazione tradizionale o animazione classica 2D, è alla base delle tecniche di animazione, ma i costi ed il notevole impegno richiesto nel tempo, sono alcune tra le ragioni per cui gli animatori hanno sviluppato metodi più efficienti. Oltre a questo, la necessità di sperimentare nuovi modi di comunicare attraverso le immagini, ha portato nel corso degli anni a sviluppare diverse tecniche di animazione che hanno subito una notevole evoluzione soprattutto grazie al progresso tecnologico, il quale ha contribuito in modo significativo nel lavoro di produzione. Si è scelto di riportare alcune tra le tecniche di animazione maggiormente impiegate allo stato odierno.

Stop Motion

La tecnica passo-uno, precedentemente nominata, prende anche il nome di stop motion. Venne introdotta ed utilizzata da Georges Méliès, che insieme ai fratelli Lumière diede origine al cinema, per la realizzazione di *Escamotage d'une dame chez Robert- Houdin* del 1896. In questo filmato, Méliès fa sedere una donna su una sedia posta al centro dell'inquadratura, dopodichè interrompe la ripresa per sostituire l'attrice con uno scheletro.

In *The VES Handbook of Visual Effects: Industry Standard VFX Practices and Procedures*[6], Pete Kozachik fornisce una descrizione della stop motion, considerandola come una mescolanza di animazione e fotografie di modelli, il mezzo più versatile per produrre gli effetti di creature. Oggigiorno, questa tecnica è scelta più per motivi stilistici, lasciando alla computer grafica il compito di produrre filmati che devono essere iperrealistici. Durante la creazione di un filmato in stop motion, un singolo animatore crea una sequenza di movimenti posizionando un manichino flessibile, appositamente costruito per il tipo di movimento desiderato, in diverse pose e fotografando ciascuna posa come un frame separato che compone l'intero video. Non disponendo dell'interpolazione tra i fotogrammi o delle procedure della fisica newtoniana, nella stop motion si evidenzia la mancanza di realismo, ulteriormente marcata dall'assenza di *motion blur*.

Per molti decenni l'animazione per questa tipologia di film si basava su un approccio lineare in cui si iniziava a posizionare il personaggio, sistemando i movimenti fino alla fine e senza poterne controllare l'andamento. Con lo sviluppo di nuovi sistemi di acquisizione hardware in grado di memorizzare centinaia di fotogrammi, gli animatori furono in grado di rivedere un'intera inquadratura in corso. Anche se tali immagini acquisite dalla videocamera non erano utilizzabili per il prodotto finale, erano invece molto utili per controllare la produzione in corso. Oggi tali prodotti sono utilizzabili grazie ad applicazioni per PC o Mac che richiedono solo una videocamera e una scheda grabber.

Il direttore della fotografia e produttore di effetti speciali, Pete Kozachik, precedentemente citato, prese parte alla realizzazione di alcuni tra i più noti film in stop-motion, come *The Nightmare Before Christmas* del 1993 e *Coraline* del 2009. Tra gli esempi più recenti vi sono *Kubo and the Two Strings* del 2016, figura 1.9, e *Isle of dogs* del 2018, in figura 1.10.

Figura 1.9: Backstage di *Kubo and the Two Strings*. Fonte: The Hollywood Reporter.

Figura 1.10: Backstage di *Isle of dogs*. Fonte: Canon.

Rotoscoping

La tecnica del *rotoscoping* fu brevettata nel 1917 da Max Fleischer e la utilizzò per la produzione di una serie di cortometraggi muti dal nome *Out of the Ink Well*. Tale metodo si basava sull'utilizzo di un proiettore che riproduceva su di una superficie le singole immagini realizzate manualmente su carta.

Allo stato odierno, nel processo di rotoscoping i filmati digitali sono caricati in un apposito pacchetto software e gli artisti di effetti visivi utilizzano delle spline per tracciare i contorni delle superfici necessarie. A seconda del tipo di software, le spline possono essere convertite in immagini opache oppure possono essere lette nel pacchetto di compositing. In alcuni casi è possibile che il software incaricato della composizione importi direttamente i file delle spline in modo da poter essere modificate.

Per ottenere un buon risultato è necessario analizzare attentamente il video su cui si intende lavorare e studiare come si articolano i personaggi presenti.
Tra gli esempi che utilizzano la tecnica del rotoscoping si possono citare il video musicale girato nel 1985 della canzone *Take on me* del gruppo norvegese A-ha il quale utilizzò un misto di live-action, schizzi a matita e rotoscoping, figura 1.11; oppure la serie animata *Undone*, in figura 1.12, che venne realizzata interamente con questa tecnica e venne lanciata su Amazon a settembre del 2019.

Figura 1.11: Scena del video musicale *Take on me*. Fonte: YouTube.

Figura 1.12: Serie animata *Undone* realizzata con il rotoscoping. Fonte: The Verge.

Motion graphics

La tecnica di *motion graphics design*, comunemente conosciuta come motion graphics consiste nell'animazione digitale di figure e soprattutto di testi.
Combinando diverse figure e testi si ottengono delle rappresentazioni visive in grado di trasmettere un messaggio. Infatti, questo tipo di animazione rappresenta

uno strumento efficace nell'ambito dello storytelling, in quanto attraverso grafiche semplici è possibile raccontare concetti anche molto complessi.

Oggigiorno la tecnica di motion graphics è notevolmente utilizzata soprattutto nel settore marketing per la realizzazione di spot pubblicitari, ma anche nei lungometraggi o video di tipo educational, nella produzione di sigle di programmi televisivi e generalmente per l'animazione di loghi e nella produzione di interfacce grafiche su diversi dispositivi.

Per realizzare delle grafiche animate efficaci occorre tenere in considerazione alcuni aspetti, quali:

- la scelta del colore: l'uso di colori dalle tonalità contrastanti possono creare effetti poco gradevoli, per cui non solo è necessaria una palette di colori adatta ma occorre anche fare attenzione al significato dei colori, scegliendoli in base al concetto che si intende trasmettere e considerando il target di riferimento.
- Il corretto utilizzo del testo: nel caso dei video, per rendere comprensibile le scritte è opportuno utilizzare una tipografia di tipo sans serif invece che serif, in quanto permette una lettura più immediata. Occorre inoltre prestare attenzione alla tipologia di font, evitando l'utilizzo di font che dispongono di caratteri poco definiti e facilmente deformabili. Inoltre, per conferire un aspetto più armonioso alla grafica del video, è preferibile limitarsi all'uso di uno o al massimo due tipi di font diversi, ed è necessario allineare adeguatamente tali elementi testuali rispetto all'inquadratura.
- L'utilizzo dei 12 principi dell'animazione: per poter creare animazioni fluide e di impatto occorre seguire accuratamente i principi fondamentali alla base di qualsiasi tecnica di animazione. Tali criteri sono descritti in modo approfondito nel quarto capitolo.

Numerosi sono i video realizzati con la tecnica della motion graphics e di questi vengono riportati due esempi visibili nelle figure 1.13 e 1.14.

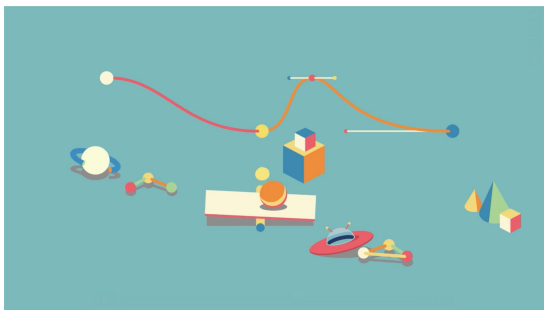


Figura 1.13: Frame di: *This is Motion Graphics!* Fonte: Vimeo/ Kevin Cheng.

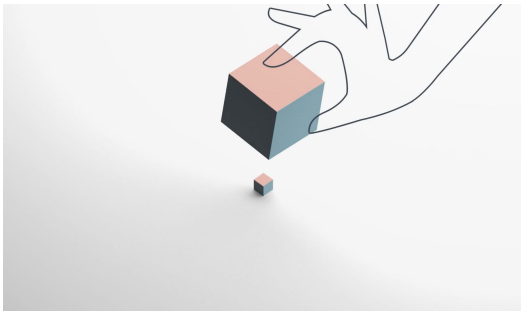


Figura 1.14: Frame di: *Are we Living in a Simulation?* Fonte: Vimeo/ Eoin Duffy.

Animazione 3D

L'animazione 3D è un tipo di tecnica che necessita della presenza del computer, nonché di software creati appositamente per svolgere i compiti richiesti dall'animatore, ossia animare oggetti solidi visibili a 360 gradi.
Come descritto nel libro *3D Animation Essentials*[7], il 1960 rappresenta l'anno emblematico della nascita della computer grafica e della computer animation, in quanto il computer che fino ad allora veniva usato unicamente come dispositivo di calcolo, divenne lo strumento fondamentale che rese possibile il cambiamento nel mondo dell'animazione; in particolare la presenza dell'hardware per l'interazione con l'utente e del software per i cambiamenti in tempo reale.

Nello specifico, la computer animation prevede il movimento di modelli tridimensionali derivanti da poligoni basilari quali, il cubo, la piramide, la sfera ed il cilindro. Tali modelli, per poter essere animati necessitano del *rigging*, ossia la creazione di una struttura da associare all'oggetto in questione nei punti in cui si desidera ricreare il movimento. Dal punto di vista estetico, i modelli presentano dei materiali e texture, scelti in base allo stile, realistico o fittizio, che si intende rappresentare.
Il vantaggio di questa tecnica di animazione, rispetto all'animazione digitale 2D è la possibilità di avere sempre presente il soggetto da animare, risparmiando così il lavoro di produzione dello stesso personaggio in pose diverse.

Il primo film di animazione realizzato nel 1995 con la computer grafica è *Toy Story*, una produzione che diede il via ad una crescita esponenziale nell'ambito della CGI (Computer-Generated Imagery). Infatti nei vent'anni successivi fino allo stato odierno si possono contare numerosi film in 3D che hanno riscontrato enorme successo, tra questi vi sono *Avatar*, *Up*, *Dragon Trainer*, *Madagascar*, *The Secret Life of Pets* e molti altri. Ma la CGI trova impiego in numerosi settori oltre al cinema, tra cui il più essenziale è il settore dei videogiochi.

Figura 1.15: *Toy Story*, John Lasseter, 1995. Fonte: Disney·Pixar.

Nonostante l'evoluzione delle tecniche di animazione permettano un lavoro di produzione più efficiente, il metodo tradizionale rimane un punto di riferimento essenziale, che i film odierni continuano a seguire, dando quindi particolare attenzione ai dettagli con l'obiettivo di creare animazioni fluide e di incuriosire lo spettatore.

Per quanto riguarda l'animazione digitale 2D, si è deciso di trattare l'argomento in maniera più approfondita nel paragrafo successivo, in quanto tale tecnica è stata utilizzata nella realizzazione del cortometraggio oggetto della tesi.

1.1 L'animazione digitale 2D

L'animazione digitale 2D nasce verso la fine degli anni '90 dalla necessità di ridurre i tempi, la complessità di lavorazione e i notevoli costi che richiede l'animazione tradizionale. Questo tipo di animazione venne definita come*flash animation* dal nome del software Adobe Flash con cui venne realizzata la prima serie animata pubblicata sul web, *The Goddamn George Liquor Program*.

Grazie al contributo tecnologico, in particolare all'uso del computer, la flash animation offre la possibilità di implementare le classiche tecniche manuali, quali sovrapposizione, *onion skinning*, *morphing* e rotoscopio, in modo da ottenere varianti più sofisticate ed efficaci dal punto di vista funzionale, economico e soprattutto tempistico. Tali operazioni non sarebbero possibili senza l'utilizzo di programmi quali Animate o After Effects della suite Adobe che rispondono alle caratteristiche tipiche dell'animazione 2D vettoriale.

Principalmente gli animatori necessitano quindi di una semplificazione nel processo di produzione e tale richiesta è resa possibile mediante l'introduzione nei software di nuove funzioni come il *puppet tool*, che consente all'animatore di controllare con più facilità il movimento del personaggio tramite il posizionamento di pin nei giunti dell'immagine; oppure attraverso lo sviluppo di nuove tecniche come l'interpolazione tra i frame. Quest'ultima tecnica consente agli animatori di limitare il numero di disegni utilizzati e di effettuare piccole variazioni come la modifica del colore o la frequenza dei fotogrammi, con tempistiche pressoché istantanee, facilitando in questo modo il processo di animazione.

Per comprendere meglio il significato dell'animazione 2D vettoriale, occorre effettuare una distinzione tra grafica raster e grafica vettoriale. La grafica raster, detta anche grafica bitmap o semplicemente bitmap, consiste in una griglia ortogonale formata da un cospicuo numero di pixel, ciascuno di colore diverso ed accostati in modo da generare un'immagine.

Quando si effettua l'operazione di ingrandimento di una bitmap, i pixel devono essere *stirati*, spesso risultando frastagliati, o interpolati, e ciò comporta una perdita di informazioni notevole che rappresenta la causa dell'effetto visivo sgranato e poco piacevole in cui i singoli pixel diventano visibili all'interno dell'immagine. Nello specifico, quando un programma di grafica interpola l'immagine, esso crea nuovi pixel, stimando il valore del colore sulla base delle informazioni dei pixel esistenti. La proprietà più importante delle immagini raster è quindi la risoluzione che può

essere espressa in due modi diversi a seconda dell'impiego.
Il primo parametro è definito DPI, o punti per pollice, e si tratta di una grandezza utilizzata esclusivamente per dispositivi fisici come le stampanti, e dal punto di vista tecnico si riferisce alla quantità di punti di inchiostro che una stampante è in grado di depositare su una linea di lunghezza pari ad un pollice, ossia 2,54 cm. Tale unità di misura viene quindi utilizzata solo nel caso in cui si necessita di stampare un'immagine digitale. Pertanto, per ottenere un risultato ad alta risoluzione si dovranno impostare i parametri adeguati a seconda del tipo di stampante, considerando che quelle a getto d'inchiostro hanno una risoluzione dai 300 ai 720 DPI, mentre nel caso delle stampanti laser si possono ottenere immagini a una risoluzione che va dai 600 ai 2.400 DPI. Il modello di colore tipico del DPI è il CMYK costituito dai colori ciano, magenta, giallo e nero, e basato sulla sintesi sottrattiva.
L'unità di misura espressa in PPI, pixel per pollice, è invece propria delle immagini digitali visualizzate sui display, ma influisce anche sul risultato di stampa come precedentemente descritto. Nello specifico, tale grandezza rappresenta la densità di pixel contenuti in un pollice, visualizzati all'interno di un'immagine digitale e quindi la quantità di dettagli che presenta tale immagine; ma indica anche il numero di pixel totali che un display può contenere. Il parametro PPI si basa sul modello di colore a sintesi additiva, RGB, definito dai colori rosso, verde e blu.
Considerando tale grandezza e dal momento che la dimensione e i dettagli dell'immagine sono definiti dalla sola densità dei pixel, per ottenere immagini ben definite, destinate unicamente alla visualizzazione sullo schermo sono sufficienti 72 PPI.
L'utilizzo della grafica raster trova risultati ottimali nei casi in cui è necessario mostrare la transizione graduale di sfumature e colori, ad esempio nelle fotografie oppure per realizzare cataloghi, volantini e altri progetti analoghi. Il software più noto e diffuso per manipolare le immagini raster e ottenere risultati ad alta qualità è sicuramente Adobe Photoshop.
I principali formati di queste immagini sono JPG, PNG, GIF e TIF.

Restando nel contesto dei pixel, è interessante riportare una campagna di sensibilizzazione dal nome *Population by Pixel*[8], realizzata nel 2007 dal WWF, in collaborazione con il designer Mikami Yoshiyuki, con l'obiettivo di riportare l'attenzione sulle specie animali a rischio di estinzione. Sono state utilizzate delle foto di alcuni di questi animali in cui ogni pixel di cui si compone l'immagine rappresenta un esemplare rimasto allo stato brado, figura 1.16.

Recentemente quest'idea è stata ripresa da un programmatore, Joshua Smith, aggiornando il numero di pixel di queste immagini con il numero esatto di esemplari rimasti in vita, sulla base delle statistiche rilasciate da Animal Planet. Tali immagini sono diventate subito virali sul web, figura 1.17.

Come si può notare, alcune di queste immagini risultano talmente sfocate da rendere l'animale in questione incomprensibile, in quanto presenta un numero di pixel,

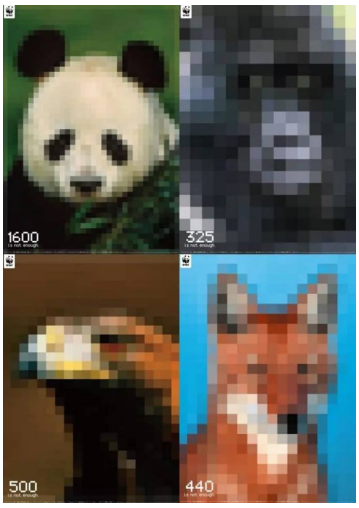


Figura 1.16: *Population by Pixel.* Fonte: NIJI

Figura 1.17: Immagini caricate da Joshua Smith sul sito Imgur. Fonte: PetaPixel.

esemplari della specie, prossimo allo zero. L'intenzione è proprio quella di mettere in evidenza quanto la situazione della biodiversità animale sia drammatica, e grazie all'utilizzo delle immagini tale concetto viene trasmesso alle persone in maniera più efficace, ottenendo un livello di attenzione superiore rispetto alla moltitudine di dati e statistiche numeriche, di più difficile comprensione.
Da un punto di vista tecnico, le immagini così rappresentate possono essere considerate degli esempi perfetti per comprendere come si manifesta un'immagine raster quando viene ingrandita; risultato che invece non si ottiene con un'immagine di tipo vettoriale.

La grafica vettoriale consente di convertire un'immagine raster in un'immagine formata da *path*, percorsi, ossia un'immagine costituita da linee o curve, vettori, definiti mediante formule matematiche. Queste indicano come sono modellati i percorsi e il colore con cui sono riempiti o delimitati.
A differenza della bitmap, la grafica vettoriale è indipendente dalla risoluzione: può essere ingrandita o rimpicciolita senza nessuna perdita di qualità e definizione, figura 1.18. Infatti durante lo scalamento, le formule matematiche che definiscono i vettori non si modificano, di conseguenza l'immagine rimane inalterata in quanto non si verifica alcuna perdita di informazioni. Il formato vettoriale è impiegato soprattutto nella realizzazione di loghi affinché questi siano ben definiti a qualsiasi ingrandimento e risoluzione e quindi utilizzabili in diversi contesti. La grafica vettoriale permette inoltre di visualizzare anche solo i contorni delle illustrazioni realizzate e questo comporta numerosi vantaggi rispetto alla grafica raster. Per quanto riguarda i formati vettoriali, quelli più comuni sono: AI, EPS, PDF e SVG.

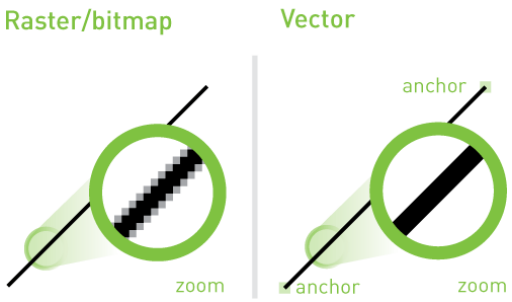


Figura 1.18: Ingrandimento applicato ad un'immagine raster e vettoriale.
Fonte: Elliots ICT Blog.

Il programma di editing grafico più noto e utilizzato per la vettorializzazione delle immagini è Adobe Illustrator, impiegato per la realizzazione di alcune illustrazioni del progetto di tesi, tra cui le grafiche esplicative dei giochi previsti da CuBee e il logo del sistema.

Un ulteriore aspetto da considerare riguarda la conversione delle immagini. Mentre la conversione dalla grafica vettoriale alla grafica raster è facilmente realizzabile, il passaggio inverso, da raster a vettoriale, è molto più complicato, se non inattuabile nella maggior parte dei casi. Per tale motivo occorre analizzare e stabilire quale tipo di grafica utilizzare prima dell'esecuzione di qualsiasi illustrazione.
Inoltre è anche possibile utilizzare una combinazione di immagini raster e vettoriali e tale scelta è stata effettuata per la produzione del cortometraggio oggetto della presente tesi, in cui le illustrazioni sono state realizzate usando i vettori in combinazione ad alcuni effetti raster per ottenere un particolare effetto visivo, che verrà descritto in seguito.

Principalmente l'animazione digitale 2D lavora con la grafica vettoriale e prevede tre fasi nel processo di realizzazione: pre-produzione, produzione e post-produzione. Generalmente una pipeline di produzione di un'animazione 2D si compone delle fasi rappresentate in figura 1.19.

Se finora sono state descritte alcune delle tipologie di animazione esistenti, nel paragrafo successivo verrà affrontata l'animazione dal punto di vista dell'impiego che offre in diversi contesti. Successivamente ci si addentrerà nella produzione effettiva del cortometraggio, nonché nell'uso dell'animazione digitale 2D.

1.2 La versatilità dell'animazione digitale

L'animazione, grazie al supporto della tecnologia ottiene importanti impieghi in diversi settori. I più giovani non sono i soli destinatari dei contenuti animati, infatti

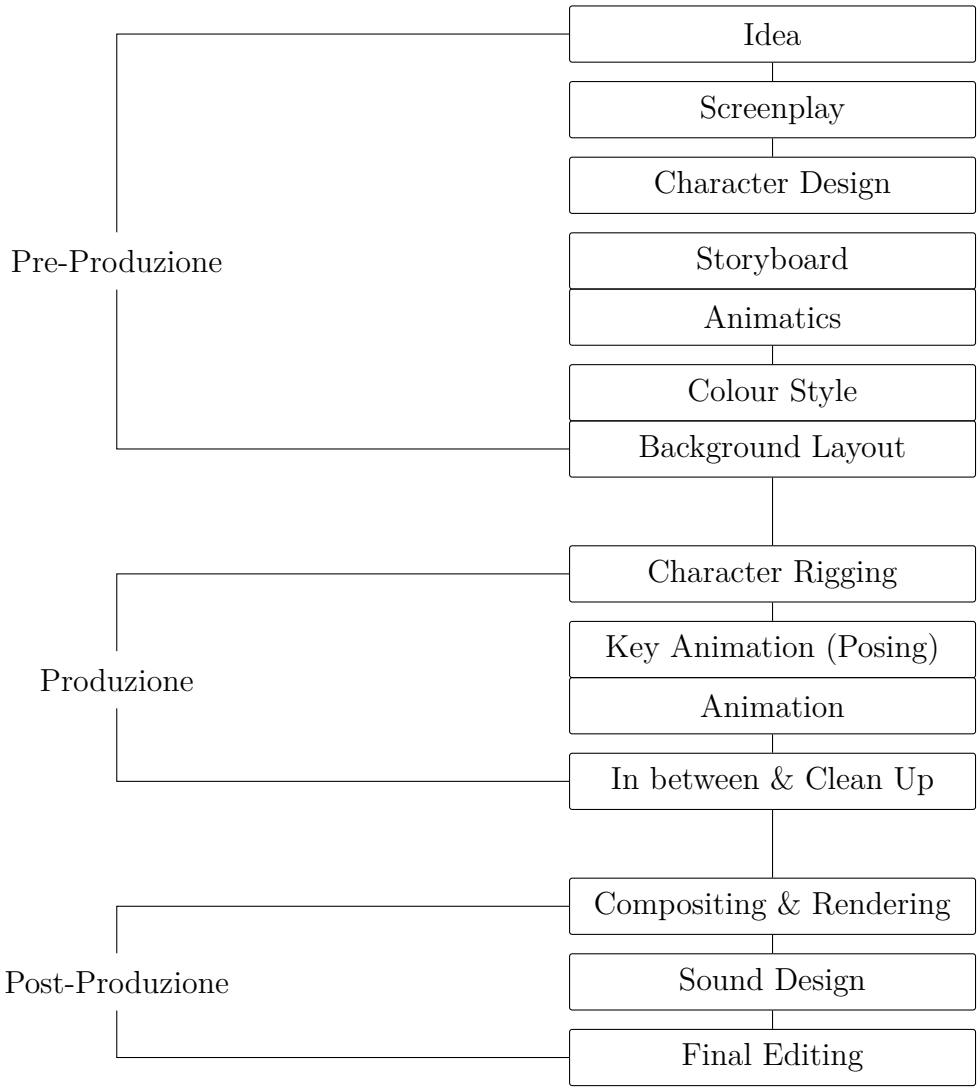


Figura 1.19: Pipeline di produzione di un'animazione

questo efficace mezzo espressivo vede sempre più ottimi risultati anche rispetto ad un target più adulto. L'animazione, oltre che nella produzione di film e ovviamente nei cartoni animati, si può quindi definire di carattere versatile nei campi indicati di seguito.

Entertainment

L'intrattenimento è il principale e il più vasto settore in cui viene utilizzata l'animazione in quanto coinvolge diverse piattaforme come tv, cinema, web e smartphone. Esistono numerose serie animate televisive create per i bambini ma anche per gli

adulti, come i classici *Tom & Jerry*, *Topolino e Paperino* o i più recenti *Rick & Morty*, *I Simpson* e *Adventure Time*. Serie tv che grazie ai software di animazione sempre più avanzati permettono di creare racconti in cui si può notare la combinazione di più stili, come nel caso del cartone animato *Lo straordinario mondo di Gumball* in figura 1.20. In questa serie sono presenti diverse tecniche di animazione quali l'animazione tradizionale, marionette, stop motion, animazione Flash e live-action. Tale tecnica mista di stili si ritrova anche nei film destinati al cinema, come *Chi ha incastrato Roger Rabbit*, *Space Jam* o i più recenti *Paddington* e *I Puffi*, in cui sono utilizzati attori reali che interagiscono con personaggi disegnati in 2D o 3D e in alcuni casi tali rappresentazioni sono così realistiche che creano un potente coinvolgimento negli spettatori. Tra gli altri numerosi film di animazione creati per il grande schermo che hanno riscontrato grande interesse da parte di un pubblico eterogeneo vi sono *Up*, *La città incantata* e il celebre *Toy Story*.

Figura 1.20: *Lo straordinario mondo di Gumball*. Fonte: Cartoon Network.

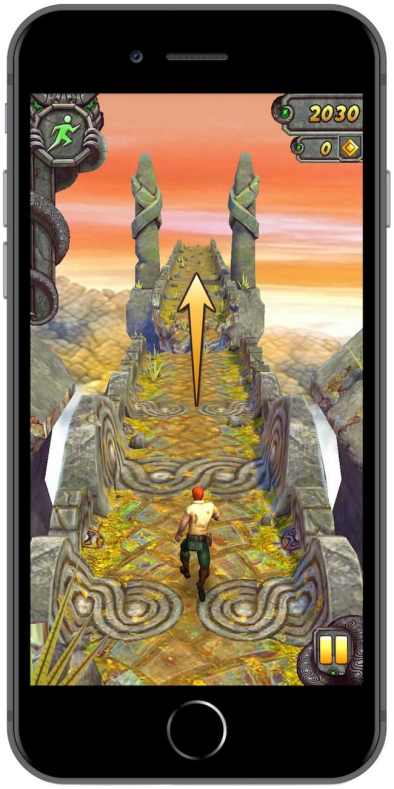


Figura 1.21: *Temple Run*. Fonte: Imangi Studios. Realizzato con MockUPhone.

La nota piattaforma online a pagamento Netflix, oltre ai film e ad altri contenuti di intrattenimento, presenta produzioni originali come le serie animate *BoJack Horseman* e *Love, Death & Robots* che hanno richiamato l'attenzione di milioni di utenti soprattutto giovani. Grazie alla creazione di applicazioni, tali contenuti diventano

disponibili anche sui dispositivi mobili, permettendo quindi un intrattenimento in qualsiasi momento.
Oltre ad applicazioni di streaming, gli smartphone offrono numerose app di intrattenimento come videogiochi dalle grafiche e animazioni accattivanti che possono occupare l'attenzione delle persone per una quantità di tempo considerevole. Tra questi vi è il popolare *Temple Run*, figura 1.21.

Gaming

Il settore dei videogiochi è quello in cui l'animazione trova maggior impiego. A differenza di quella cinematografica, l'animazione in un videogioco è attiva, ossia comunica ed interagisce continuamente con il giocatore, attraverso feedback ai comandi dell'utente.

Figura 1.22: Screenshot dal videogioco Halo. Fonte: reddit/u/BlindFireSniper.

L'animazione di personaggi e di effetti visivi presenti in un videogioco, conferisce a tale medium un enorme potere di coinvolgimento per le persone di qualsiasi fascia d'età, tanto da rendere questo linguaggio un fenomeno culturale di massa. Ciò che permette una simile connessione tra utente e videogioco è proprio la caratteristica di quest'ultimo di rispondere alle volontà interattive e visive richieste dal giocatore. L'utente infatti, in alcune tipologie di gioco, come nei giochi di ruolo *FPS*, First Person Shooter, ha la possibilità di intervenire in prima persona sullo sviluppo della storia presentata, immedesimandosi nei personaggi a tal punto da amplificare la sensazione della propria presenza, all'interno di un contesto che resta di fatto virtuale. Un esempio, visibile in figura 1.22, è la serie di videogiochi *Halo* realizzata da *Bungie Studios*.

Con il progresso tecnologico, nonché lo sviluppo di nuove tecniche di animazione, il settore dei videogiochi continua ad aumentare la propria complessità e di conseguenza aumentano le aspettative dei giocatori che richiedono esperienze sempre più realistiche. Si può quindi affermare che l'assenza dell'animazione nel settore dei videogiochi implicherebbe l'inesistenza del settore stesso.

Education

L'uomo ha la tendenza a ricordare meglio contenuti visivi, rispetto a contenuti puramente testuali. Al giorno d'oggi, gli insegnanti prediligono l'uso dei video animati per attirare l'attenzione degli studenti in quanto contribuisce a rendere più chiari e concreti concetti di diversi argomenti che altrimenti possono risultare più difficili da apprendere, come nel caso di un testo scritto. Infatti, il nostro cervello è strutturato per ricevere ed elaborare input visivi più di quanto avviene con le informazioni di tipo testuale. Di seguito è illustrato un esempio di animazione di tipo educativo, figura 1.23.

Figura 1.23: Frame di un video animato in 2D utilizzato per spiegare la matematica ai bambini in modo divertente. Fonte: 3D-Ace.

Secondo molteplici studi in merito, una scena o una situazione visiva richiede solo 100 ms per essere compresa, mentre l'elaborazione di un simbolo e l'attribuzione di un significato avviene in circa 250 ms. Quindi l'uso di un formato grafico è particolarmente efficace per l'illustrazione di concetti complessi e al contempo catturare l'attenzione e l'interesse. Al contrario dei testi scritti, le immagini vengono

archiviate facilmente dal cervello nella memoria a lungo termine e sono quindi più adatte a veicolare contenuti che devono essere ricordati nel tempo.

Un ulteriore aspetto di un video animato da considerare riguarda il suo alto potenziale emozionale che permette quindi di creare una connessione emotiva con l'utente.
A livello cognitivo, il cervello *legge* un contenuto video sessantamila volte più rapidamente rispetto a un contenuto di testo. Inoltre guardare un video non richiede una partecipazione cerebrale attiva perciò l'elaborazione del contenuto non necessita di grandi quantità di energie[9].
A tal proposito, è interessante riportare alcune parti di un'intervista [10] al Prof. Francesco Giordano, neuropsichiatra infantile, psicoterapeuta, psicoanalista e docente presso l'Istituto di Specializzazione in Psicoterapia Individuale e di Gruppo (SPIGA) di Roma, in cui si esprime riguardo a come il cervello reagisce all'osservazione di un'immagine:

> "Per l'immagine, che non prevede una consequenzialità nell'attenzione come accade invece per la lettura di un testo, l'approccio a essa è 'globale', ancora più correlato, in questo caso, alla funzione e all'espressione del cosiddetto 'cervello emozionale'. Nei confronti di un'immagine, ad esempio un'opera artistica, non c'è una pianificazione nel dare attenzione prima a delle parti e poi ad altre, almeno in un primo momento. Un esame più dettagliato, come quello compiuto da un esperto, rappresenta una fase successiva nella quale vengono chiamate in causa ben precise funzioni cognitive, finalizzate anche a una valutazione dello stile dell'opera."

Mettendo a confronto l'operazione di lettura del testo e di osservazione delle immagini da un punto di vista della capacità cognitiva generale, il Prof. Giordano afferma:

> "Nella lettura di un testo è coinvolta essenzialmente una forma d'intelligenza più evoluta, 'sequenziale', la quale richiede una successione rigorosa che articola e analizza i codici grafici disposti in linea; vengono trattate più informazioni, ma stabilendo una successione, una gerarchia, un ordine, quindi. Quando guardiamo un'immagine, invece, un quadro o un'immagine pubblicitaria, utilizziamo principalmente un'intelligenza 'simultanea', dove non è possibile dire – in un primo momento – cosa guardiamo prima e cosa guardiamo dopo."

Stage Shows

L'integrazione dell'animazione negli spettacoli teatrali crea un'atmosfera più coinvolgente per lo spettatore. È infatti possibile realizzare effetti visivi che catturano l'attenzione e ricreare tutto ciò che si immagina o che altrimenti non sarebbe possibile realizzare. Oggigiorno infatti, molti produttori di spettacoli teatrali tendono a

sperimentare nuovi modi per creare scenografie che, grazie alla tecnologia in combinazione con l'animazione, creano un'atmosfera diversa dalle originarie e classiche rappresentazioni che non disponevano di tali contributi.
Sono molte le sperimentazioni teatrali che utilizzando l'animazione nelle più svariate forme, da quella digitale a quella che impiega parti meccaniche in movimento. Tra queste vi è quello della compagnia britannica *1927*, fondata nel 2005 dall'illustratore e animatore Paul Barrit e dalla scrittrice, attrice e regista Suzanne Andrade, che ha portato in scena lo spettacolo teatrale intitolato *Golem*, in figura 1.24.

Figura 1.24: Estratto di una scena dello spettacolo teatrale *Golem*.
Fonte: Bernhard Miller/ The Sydney Morning Herald.

Un'opera teatrale, arrivata anche al Teatro Stabile di Torino, che si interroga sul rapporto tra uomo e tecnologia come si evince dal titolo stesso dell'opera: golem dalla parola ebraica *gelem* significa materia grezza o robot[11].

Si tratta di una rappresentazione che unisce la dimensione fisica degli attori con la dimensione digitale delle grafiche proiettate che non fanno da fondale ma accompagnano i performer diventando parte integrante della storia e creano così uno stile narrativo unico che simula le *graphic novel*.

In un articolo del *The Sydney Morning Herald* [12], il produttore della compagnia 1927, Jo Crowley afferma:

> "Quando creiamo uno spettacolo, essenzialmente stiamo realizzando un film di

animazione fatto a mano e allo stesso tempo stiamo inventando uno spettacolo teatrale dal vivo. Le due cose sono sviluppate allo stesso tempo. È una cosa in continua evoluzione. Copione, immagini ed estetica visiva si evolvono contemporaneamente."

Marketing & Advertising

L'animazione è ampiamente utilizzata in questi settori poiché permette di raggiungere più persone e coinvolge maggiormente i potenziali clienti, rispetto alle consuete campagne pubblicitarie. Nel campo dell'advertising, l'animazione aiuta a spiegare le caratteristiche del prodotto o del servizio arrivando rapidamente al concetto chiave ed il vantaggio nell'utilizzare tale linguaggio espressivo consiste nel rendere originale il video promozionale, catturando così l'attenzione degli spettatori che saranno eventualmente più propensi ad acquistare il prodotto.

Figura 1.25: Immagine tratta dallo spot pubblicitario *Miel Pops*. Fonte: Kellogg's.

Un esempio è la nota pubblicità dei cereali *Miel Pops*, della *Kellogg's*, figura 1.25; uno spot realizzato interamente con modelli di personaggi e animazioni in 3D, in cui per presentare le palline di mais e gli anelli ai quattro cereali al gusto miele si vedono una serie di api che a ritmo di danza descrivono le caratteristiche di tali cereali, come la croccantezza e l'ingrandimento delle palline una volta immerse nel latte. Si può quindi considerare un metodo efficace per richiamare l'attenzione, come continuano a dimostrare i sempre più numerosi spot che usano tecniche di animazione.

Medicine & Science

Nel campo medico e più in generale nel campo scientifico, l'animazione è usata per realizzare visualizzazioni accurate e rappresentative ai fini della ricerca e degli studi analitici. Tramite l'animazione si ha la possibilità di creare render di modelli 3D realistici di oggetti che garantiscono una visione più dettagliata, nonché un'analisi più efficace. Le illustrazioni mediche 3D costituiscono un rilevante vantaggio nell'istruzione e nella formazione di medici, in quanto permettono di semplificare notevolmente concetti di difficile comprensione.
Le radiografie al computer sono un esempio di tali modelli tridimensionali, tramite cui i medici sono in grado di visionare parti anatomiche in modo immediato e definire una soluzione più rapida ai problemi che si presentano. Ad esempio, l'uso di tali modelli tridimensionali animati consentono ai chirurghi di osservare le repliche del corpo umano ed effettuare simulazioni prima di condurre interventi chirurgici.

Un'applicazione medica insolita per l'animazione riguarda il gioco *Top Gun*[13], per addestrare i chirurghi e migliorare le loro capacità di coordinazione occhio-mano, prima di procedere con gli interventi veri e propri. Nella figura 1.26 sono riportati due esempi.

Figura 1.26: Immagini tratte dalla competizione *Top Gun Surgeon* tenutasi a Baltimora ad aprile 2019. Fonte: Twitter/TopGunSurgeon.

Il Dr. James Rosser del Beth Israel Medical Center, tramite la creazione di tale videogioco ha dimostrato che i chirurghi che si esercitano con tali tipologie di giochi in media almeno tre ore alla settimana compiono circa il 37% in meno di errori nella chirurgia laparoscopica, e svolgono i loro compiti il 27% più velocemente rispetto ai chirurghi non giocatori [14].
Si può quindi affermare che l'utilizzo dell'animazione, insieme allo sviluppo di nuove tecnologie, apporta un sostanziale contributo a migliorare e accrescere lo studio in tali settori, medicale e scientifico, ottenendo molteplici benefici.

Forensic Science

Nel settore della criminalistica, l'uso dell'animazione può costituire un elemento di aiuto per le ricerche effettuate dagli investigatori, ad esempio comprendendo il modo in cui gli individui coinvolti nel caso hanno subito i danni, o nel convincere una giuria dell'innocenza del loro cliente. Oggigiorno, l'animazione forense, di cui vi è un esempio in figura 1.27, sta assumendo un'importanza talmente rilevante che molti tribunali hanno dovuto realizzare una nuova posizione di lavoro; l'esperto di animazione di corte, ossia una persona qualificata per verificare la veridicità dell'animazione presentata in tribunale, in modo da assicurarsi che gli avvocati non usino immagini generate al computer per sviare la verità a proprio favore.

Figura 1.27: Ricostruzione di una scena del crimine con la collaborazione di testimoni oculari, realizzata con l'animazione 3D. Fonte: Eyewitness Animations.

L'animazione, essendo un potente strumento di comunicazione, inconsciamente pone molte persone in uno stato suggestivo e richiede la cosiddetta *sospensione dell'incredulità*, ossia una sorta di patto implicito che lo spettatore sottoscrive quando accetta la realtà animata ignorando momentaneamente le incongruenze, per potersi così immergere appieno nella storia di finzione.
L'espressione *suspension of disbelief* venne coniata dal poeta romantico Samuel Taylor Coleridge nella sua opera *Biographia Literaria* del 1817:

> "[...] venne accettato, che i miei cimenti dovevano indirizzarsi a persone e personaggi soprannaturali, o almeno romantici, ed anche a trasferire dalla nostra intima natura un interesse umano e una parvenza di verità sufficiente a procurare

> per queste ombre dell'immaginazione quella volontaria sospensione dell'incredulità momentanea, che costituisce la fede poetica."

D'altra parte però, la narrazione deve avere per oggetto una storia verosimile, caratterizzata da coerenza e credibilità. Pertanto, occorre seguire delle regole e fornire allo spettatore una costruzione dell'ambiente, *worldbuilding*, semplice, chiara e immediatamente comprensibile, curare tutti i dettagli e gli elementi in modo serio e meticoloso, altrimenti si avrà una rottura del *patto* stipulato con lo spettatore.

> "Lo scrittore fantastico deve porre attenzione alla presentazione di un mondo, impegnarsi a costruire un universo, prenderlo sul serio e approfondirne ogni suo aspetto."
>
> – *J.R.R. Tolkien*

Capitolo 2

Le fasi di lavoro: Pre produzione

— Elisa Musolino

Nei paragrafi seguenti verranno descritte le prime fasi necessarie alla preparazione del cortometraggio di animazione, nonché precedenti all'esecuzione effettiva. Tali fasi riguardano l'ideazione della storia, la stesura del soggetto e della sceneggiatura, l'analisi e lo studio dei personaggi e delle ambientazioni, l'illustrazione della storia tramite lo storyboard e una prima valutazione tempistica della durata del racconto, ossia l'animatic.

2.1 L'idea del cortometraggio

La lotta ai cambiamenti climatici è ormai da anni al centro dell'attenzione mediatica, a volte talmente enfatizzata dai media che la notizia in sé più che allarmante, risulta oggetto di disinteresse o coinvolgimento temporaneo.

L'intenzione di questo progetto è quella di creare un tipo di comunicazione che permetta di porre l'attenzione su un tema apparentemente poco interessante, trasmettere informazioni scientifiche attraverso l'uso di linguaggi semplici e strumenti diversi dal solito per poter stimolare una consapevolezza collettiva, che è uno degli aspetti fondamentali per facilitare il coinvolgimento degli utenti e poter quindi dare il via a dei cambiamenti significativi.

È un fatto scientificamente provato che il clima non solo ha immediate conseguenze ambientali, ma anche pesanti ripercussioni su numerosi fattori a livello globale, di

cui l'uomo ne rappresenta il principale responsabile. Si evidenziano infatti significative implicazioni nell'ambito dei cosiddetti Paesi in via di sviluppo, in quanto tali popolazioni, che dipendono fortemente dall'ecosistema in cui vivono, dispongono di scarse risorse per poter fronteggiare questi cambiamenti.
Un notevole impatto si verifica anche sulla salute degli esseri umani. Secondo le statistiche riportate dal National Center for Biotechnology Information del 2016 [1], l'aumento della temperatura di 1 °C ha comportato un incremento della mortalità cardiovascolare del 3,44%, della mortalità respiratoria del 3,60% e di quella cerebrovascolare dell'1,40 %. Di seguito vengono riportati alcuni dati presentati dal Ministero dell'Ambiente e della Tutela del Territorio e del Mare in [2], riguardo la situazione climatica degli ultimi anni in Italia.
Mediante la disponibilità di circa 1.100 stazioni di monitoraggio e l'analisi dei dati forniti da tali indicatori, si è stimato l'andamento del clima per quanto riguarda l'Italia nell'anno 2017. Mentre a livello globale, il 2017 si è registrato come il terzo anno più caldo della serie storica dopo il 2016 e il 2015, con un aumento della temperatura media di +1,20 °C, in Italia tale valore cresce fino a +1,30 °C con picchi in primavera (+2,0 °C) e in estate (+2,8 °C). Una situazione simile si evidenzia nel valore della temperatura superficiale dei mari italiani: sulla base dei dati riportati dalla National Oceanic and Atmospheric Administration (NOAA), rispetto al periodo 1961-1990, nel 2017 si è registrato un incremento medio di +0,84 °C.
Tali cambiamenti climatici implicano evidenti conseguenze anche a livello sociale ed economico, in particolare per i settori maggiormente colpiti quali agricoltura, silvicoltura, energia e turismo. In base a quanto riportato sul sito ufficiale della Commissione Europea [3], si stima che le sole alluvioni verificatesi tra il 1980 e il 2011, che colpirono più di cinque milioni e mezzo di persone, hanno portato a perdite economiche di oltre novanta miliardi di euro.
Un ulteriore ambito che risente di tali alterazioni climatiche riguarda la biodiversità animale e vegetale, in particolare alcune specie trovano difficoltoso l'adattamento nelle nuove condizioni createsi nei loro habitat nativi, ragion per cui esse sono esposte a maggior rischio estinzione.

Quello dei problemi ambientali è un tema vasto da trattare, pertanto in questo progetto si è deciso di soffermarsi su alcuni aspetti di tali problematiche, quali: l'inquinamento atmosferico, l'uso dei pesticidi in agricoltura, l'inquinamento da rifiuti di plastica e non solo, il riscaldamento delle acque e lo scioglimento dei ghiacciai. Problemi che si è pensato di affrontare dal punto di vista dell'impatto che provocano sulla biodiversità animale, garante della vita sulla terra. In particolare si è voluto mostrare una panoramica generale sulle condizioni di vita in cui si trovano alcune specie animali, illustrandone i corrispettivi habitat.

Gli organismi viventi, grazie alle loro differenti caratteristiche genetiche sono fondamentali per mantenere l'equilibrio tra i vari ecosistemi presenti sulla Terra. Un equilibrio che però, per svolgere un buon funzionamento necessita di una costante

attenzione verso tali diversità insite in ciascuna specie e che pertanto non dovrebbero subire alterazioni o ancor più estinguersi.
Secondo quanto riportato dal WWF, nota organizzazione internazionale non governativa di protezione ambientale, l'attuale tasso di perdita della biodiversità mondiale ha raggiunto oggi livelli "senza precedenti"[4]. Questo termine è stato utilizzato dall'IPBES, nel rapporto *Global Assessment Report on Biodiversity and Ecosystem Services* del 2019 [5], per dare una descrizione della continua e sempre più intensificata azione distruttiva dell'uomo sulla natura. In tale rapporto si è stimato come il tasso globale di estinzione delle specie si è trovato a un livello che ha superato dalle decine alle centinaia di volte la media delle estinzioni verificatesi negli ultimi dieci milioni di anni. In particolare, tramite diversi calcoli e proporzioni si è valutato che su una totalità di otto milioni di specie esistenti, tra animali e vegetali, circa un milione sono a rischio estinzione.

Tra le cause principali del declino degli habitat per opera dell'intervento umano, vi sono il sovrasfruttamento e l'agricoltura. Oltre ai cambiamenti climatici, altri fattori che costituiscono una minaccia sempre più frequente per la biodiversità riguardano la pratica della caccia, l'eccessiva pesca industriale che ha portato all'impoverimento delle specie acquatiche presenti negli oceani a cui si aggiunge l'inquinamento dovuto ai derivati del petrolio come la plastica. Ulteriori fonti di pressione riguardano l'inquinamento proveniente dalle attività minerarie e dalle infrastrutture di trasporto e l'introduzione accidentale o intenzionale da parte dell'uomo di specie aliene, ossia animali e vegetali che trovandosi in habitat diversi da quelli di provenienza, entrano in contrasto con le specie autoctone, provocando l'alterazione dell'equilibrio degli ecosistemi.

Preservare la natura, il cui fulcro è la biodiversità, significa salvaguardare la principale fonte di risorse, beni e servizi che garantiscono il benessere umano. Infatti, in base al *Living Planet Report* del 2018 [6], documento che registra lo stato di biodiversità globale, pubblicato dal WWF, si stima che a livello globale la natura offra servizi per le società umane prese in considerazione, di un valore che ammonta intorno ai 125.000 miliardi di dollari annui, una cifra superiore al prodotto interno lordo di tutti i paesi del mondo che si aggira intorno agli 80.000 miliardi di dollari. L'impoverimento della biodiversità comporterebbe quindi perdite significative sia a livello sociale che a livello economico.
La biodiversità, animale e vegetale, costituisce quindi un ruolo fondamentale per il nutrimento e per la salute di tutta la popolazione mondiale, fornisce inoltre materie prime necessarie alla produzione di energia, fibre tessili e medicinali.

Un particolare riguardo è necessario rivolgere alle specie animali appartenenti alla categoria degli impollinatori (api, vespe, farfalle, mosche, uccelli e pipistrelli), in quanto costituiscono oltre il 75% delle colture alimentari destinate al fabbisogno umano, e in termini economici si stima che il valore dell'impollinazione si aggira

tra i 235 e i 577 miliardi di dollari l'anno per i soli coltivatori.

Le api sono tra i principali, nonché tra i più conosciuti impollinatori esistenti e risultano essenziali per l'impollinazione di oltre 130 mila piante, come riportato nell'opuscolo *Alla scoperta della biodiversità* [7] redatto dall'Istituto per la Protezione e la Ricerca Ambientale, ISPRA.
Purtroppo, tali insetti stanno subendo un declino drammatico e sempre più crescente negli ultimi anni, per via delle minacce descritte in precedenza, ma nello specifico a causa della degradazione degli habitat dovuta all'intensificazione agricola, la presenza di alcune malattie, e soprattutto l'uso di antiparassitari e di erbicidi in agricoltura. Alcune ricerche attualmente in corso, ipotizzano anche un'influenza delle onde elettromagnetiche, sempre più in aumento per via dei ripetitori di telefonia mobile: si sostiene infatti che tali radiazioni interferiscano con il sistema di orientamento degli insetti, impedendo loro di rintracciare la via dell'arnia e portandoli a disperdersi e morire altrove [8].

"Se le api scomparissero dalla faccia della Terra, all'umanità resterebbero solo quattro anni di vita" è una citazione attribuita erroneamente ad Albert Einstein, ma che va tuttavia considerata in quanto delle cento colture che costituiscono il 90% della produzione mondiale di cibo, ben settantuno sono legate al lavoro di impollinazione delle api. Tali insetti sono infatti responsabili di almeno 95 specie di frutta e verdura ed è da essi che deriva la produzione del mondo vegetale grazie al quale respiriamo, visto l'enorme contributo di ricambio di ossigeno che deriva dalla fotosintesi clorofilliana delle piante. Dal lavoro di impollinazione delle api deriva inoltre la produzione di molti farmaci di uso comune. Secondo una stima dell'Organizzazione Mondiale della Sanità, il 25% dei farmaci attualmente utilizzati, tra cui l'aspirina, deriva da specie vegetali [9].

Le api possono essere considerate delle vere e proprie sentinelle dello stato di salute dell'ambiente; sono infatti estremamente sensibili alle molecole chimiche contenute nei pesticidi, una delle principali cause della loro scomparsa.
È proprio quest'ultimo aspetto, ossia il riconoscere nella figura di questi insetti una sorta di *campanello di allarme* riguardo i problemi ambientali, che ha scaturito l'idea di attribuire all'ape un ruolo da protagonista.
Nello specifico, nel cortometraggio si è deciso di usare la figura dell'ape come guida per l'utente. Sarà proprio lei, insieme ad un cubo, che intraprenderà un viaggio tra diversi ecosistemi per raccogliere le adesioni di alcuni degli animali più in pericolo ad unirsi ad uno sciopero collettivo del regno animale nato proprio nell'ecosistema delle api, ormai giunte al limite della sopportazione dei danni causati dall'uomo.

Il cubo, quale elemento di sinergia con le api, rappresenta il mezzo attraverso cui visionare i pericoli ambientali captati dalle api stesse. Nell'ecosistema delle api raffigurato, questo dispositivo può essere definito come un centro di controllo dello stato della fauna terrestre.

Nell'esperienza proposta all'utente, il cubo ha inoltre un ruolo fondamentale in quanto viene presentato come elemento materializzato quindi tangibile, attraverso cui l'utente può interagire in prima persona e dare un proprio contributo durante il viaggio intrapreso dall'ape protagonista. L'interazione tra il cortometraggio e il cubo come oggetto fisico si propone di creare un miglior senso di immersione nell'esperienza, rendendo l'utente protagonista stesso della storia. Il ruolo che ha l'utente all'interno del sistema CuBee e lo svolgimento dell'esperienza verrà trattato in dettaglio nella terza parte del documento.

Per motivi tempistici dovuti alla complessità del progetto, si è deciso di soffermarsi su tre ambienti: il cosiddetto mondo delle api, ovvero l'ambiente rurale da cui hanno origine gli alimenti fondamentali per l'uomo e in cui le presenza delle api è indispensabile; l'ambiente marino e l'ambiente artico. Questi mondi sono illustrati dal punto di vista di alcuni degli animali più noti e maggiormente coinvolti. Attraverso questi animali si è voluto mostrare, sebbene in un contesto di fiction, ma ispirato a situazioni reali, lo stato in cui essi sono costretti a vivere e gli effetti che l'uomo stesso sta subendo per via della superficialità posta rispetto a tali problemi.

Per delineare la storia, inizialmente è stata svolta una fase di ricerca sullo stato dell'arte riguardo i problemi ambientali, sono stati raccolti dati, documenti e libri che mostrano i cambiamenti nel tempo, soffermandoci sui dati più attuali.
Siccome la storia narrata nel cortometraggio ha come fulcro gli animali, è stato effettuato uno studio per comprendere meglio il comportamento degli animali presi in considerazione, i rischi di estinzione che essi corrono e il motivo per cui la loro presenza è fondamentale nell'ecosistema in cui viviamo.
Si è voluto quindi rappresentare una situazione inversa, raccontata dal punto di vista degli animali: cosa accadrebbe al nostro pianeta se tutti gli animali decidessero di scioperare?

2.2 Soggetto e sceneggiatura

Il soggetto è il primo passo da affrontare per realizzare una storia. Consiste in una breve sintesi di quanto verrà rappresentato, ossia la presentazione delle ambientazioni, la contestualizzazione temporale, le caratteristiche dei personaggi e gli sviluppi della narrazione. Dal punto di vista strutturale, il soggetto è diviso in tre parti: antefatto, sviluppo e conclusione.
Occorre innanzitutto definire l'idea nella fase di brainstorming e per fare ciò è necessario partire dal tema che si desidera trattare, informandosi ed ispirandosi a dei validi esempi narrativi. Per poter ottenere un risultato efficace e quindi coinvolgere il destinatario della lettura del soggetto, durante la fase di scrittura, occorre tenere in considerazione alcuni aspetti fondamentali:

- il metodo di scrittura deve essere accurato, non bisogna riportare concetti astratti come ad esempio il pensiero e le emozioni, ma piuttosto spiegare le espressioni ed i gesti che il personaggio compie per rappresentare tali azioni astratte. Occorre quindi scrivere per immagini, creando una successione di eventi che compongono l'intera storia e che permettono al lettore di interpretare e seguire in modo fluido l'andamento delle vicende.
- L'articolazione delle frasi non deve essere complessa, pertanto occorre semplicità ma allo stesso tempo chiarezza lessicale. Le proposizioni devono quindi essere dirette ed avere una lunghezza molto contenuta.
- Per quanto riguarda il tempo verbale da usare, occorre utilizzare l'indicativo presente coniugato alla terza persona, in quanto consente al lettore di immedesimarsi maggiormente nella lettura e allo stesso tempo evita all'autore del soggetto di intervenire con commenti personali sulla descrizione degli eventi.

Il cortometraggio realizzato per la tesi prevede un'unica storia articolata in quattro parti che hanno come filo conduttore la protagonista Pinnie, intenta ad attraversare tre ambientazioni diverse, ed il cubo, quale oggetto, virtuale e tangibile, è di fondamentale importanza nell'evoluzione della storia.
Tale suddivisione è stata dettata dalle necessità progettuali del sistema CuBee, un sistema di storytelling interattivo il quale combina la proiezione di contenuti visivi, narrativi ed informativi, con l'uso di dispositivi fisici, i cubi, che agiscono attivamente nei punti in cui la narrazione richiede il coinvolgimento e l'azione diretta da parte dell'utente; azioni che verranno trattate in dettaglio nella terza parte del documento, ossia la dimensione interattiva.

Per una maggior chiarezza riguardo l'andamento della storia e per evidenziare i punti in cui è presente l'intervento dell'utente sulla narrazione stessa, di seguito è descritto il soggetto per ciascuna delle quattro parti del cortometraggio.

Prima parte

In un campo di fiori e pomodorini, tre api sono intente ad impollinare. All'improvviso, una nube tossica invade le api.
Due api infermiere trasportano sulle barelle due delle api intossicate dalla nube verso il reparto pesticidi. Nella sala di produzione del miele, le api operaie lavorano il miele, mentre Regina usa il cubo, dispositivo di controllo, per guardare e tenersi aggiornata sulle notizie principali riguardo le condizioni in cui vivono gli animali coinvolti nei cambiamenti climatici.

Seconda parte

Un forte rumore proveniente dalla porta di ingresso fa sussultare Regina e le api presenti nel laboratorio. Dalla porta entra Pinnie che gesticola affannata spiegando

a Regina l'ennesimo caso di moria delle api a causa dei pesticidi.
Il giorno seguente, Pinnie, con la tracolla indosso, è fuori dalla porta del loro centro operativo, mentre tutte le api, munite di bagagli, escono e volano via. L'ultima ad uscire è Regina che voltandosi verso Pinnie, le consegna il cubo.
Pinnie attacca un cartello alla porta d'ingresso chiusa dalle serrande e vola via. Sul cartello è visibile la scritta "Strike".
Un apicoltore davanti al centro operativo delle api, guarda confuso il cartello, grattandosi il capo.

Pinnie si trova su di una spiaggia. Cammina verso l'oceano ed entra in acqua.
I fondali sono coperti di rifiuti, i coralli hanno colori sbiaditi e le alghe stanno sudando. Lungo la strada compare il cartello "Fishadelphia", seguito dal cartellone che preannuncia il pericolo di una tempesta di rifiuti.
Nell'asilo, tre pesciolini guardano attentamente Mrs. Cuttlefish, la maestra seppia che spiega alla lavagna i pericoli sott'acqua introdotti dall'uomo. All'improvviso un forte boato fa sobbalzare i pesci dal pavimento dell'aula, i fondali diventano bui ed inizia una raffica di rifiuti che invadono i fondali. Pinnie si ripara spaventata sotto una conchiglia.
La tempesta è finita. Fish, un pesce dalla lunga coda spazza i rifiuti davanti alla propria casa affiancandoli all'enorme montagna di rifiuti accumulati.
Vicino ad un semaforo, il polipo Octopus, con il giubbotto catarifrangente, distribuisce i giornali. Pinne si avvicina a lui, gli consegna un volantino su cui è visibile la scritta "Join the Strike", e nuota via. Octopus guarda il volantino e accenna un sorriso finale.
Pinnie nuota tra le case dei pesci ormai distrutte. Si sentono delle urla di aiuto. Pinnie si dirige velocemente nella direzione da cui provengono quei suoni.
La tartaruga Turtle, intrappolata in una rete da pesca, agita le zampe per liberarsi. Pinnie, si avvicina e con l'aiuto del suo pungiglione, taglia la rete e libera la tartaruga. Pinnie in groppa a Turtle nuotano verso la superficie dell'oceano.

Terza parte

Pinnie è seduta su di un contenitore di plastica galleggiante sulla superficie dell' oceano. Un pescatore, seduto sul pontile, tiene tra le mani una canna da pesca. Le spiagge dell'isola sono ricoperte di rifiuti ed un capodoglio spiaggiato giace tra questi. Pinnie tiene lo sguardo abbassato. All'improvviso il gabbiano Gull si posa bruscamente sul contenitore facendo cadere Pinnie in acqua. Pinnie risale in superficie fissando infastidita Gull, che posa davanti a sé il mozzicone di sigaretta che tiene tra il becco.
Il pescatore agitandosi, si sforza a tirare su la sua lenza che si muove freneticamente. Egli alza la canna da pesca e vede un gommone di una ruota attaccata all'amo. Un delfino guarda la scena sorridendo. Pinnie in groppa a Gull volano via.

Tra i ghiacciai, Pinnie è seduta tremante sulla schiena di Bear, un orso polare sdraiato sulla superficie ghiacciata. Un ventilatore accanto a loro emana quest'aria fredda.
Bear si alza facendo scivolare Pinnie giù dalla sua schiena. Lo stomaco dell'orso brontola. Egli cammina verso un piccolo lago tra i ghiacciai sulla cui superficie sono visibili chiazze nere di petrolio. Accanto vi è un secondo orso intento a cercare cibo in un barile di petrolio.
Il suono di un aereo attira l'attenzione dei due orsi che guardano in quella direzione. Dall'aereo atterrato, South pole airlines, scendono tre pinguini muniti di cappelli, sciarpe e tavola da snowboard. I tre pinguini si dirigono verso Orso.
Il capo pinguino tiene nella pinna un volantino del resort "Brrland" e lo alza verso Orso per chiedergli informazioni. Bear, con la bocca aperta e lo sguardo incredulo, alza la zampa indicando la direzione ai pinguini. Questi guardano ripetutamente il volantino e la zona indicata che ora è distrutta. I pinguini risalgono velocemente nell'aereo e ripartono.
Pinnie mostra a Bear il volantino dello sciopero e gli chiede l'adesione. All'improvviso si sentono delle urla provenienti da un cucciolo di orso che si trova su di un ghiacciaio staccatosi e che si sta allontanando da Bear e Pinnie.

Quarta parte

Bear firma l'adesione allo sciopero. Una petroliera si fa largo tra i ghiacciai lasciando dietro di sé una voluminosa nube di fumo nero.

Su di una piccola isola incontaminata sono riuniti tutti gli animali che hanno aderito allo sciopero. Osservano attentamente Pinnie, in piedi sul capo di un suricato, che tiene in mano il volantino con le richieste da fare agli uomini per fermare i cambiamenti climatici.

Si sentono schiamazzi di bambini che giocano in un parco. Sul prato è adagiato il volantino con le richieste degli animali. Un bambino nota il volantino, lo raccoglie e lo legge. Si sente il rintocco di un campanile con un orologio che segna le ore 12:00. Il bambino guarda l'orologio, poi il volantino ed inizia a correre verso destra uscendo dall'inquadratura.

Sceneggiatura

La stesura della sceneggiatura corrisponde alla rappresentazione dettagliata dell'idea che si intende comunicare. Ovvero consiste nell'articolare il racconto suddividendolo in scene, ciascuna delle quali contenente le ambientazioni ed i riferimenti temporali in cui si svolge la storia, le azioni specifiche compiute dai personaggi, gli effetti sonori e gli eventuali dialoghi. Nello specifico, dal punto di vista narrativo,

una storia deve tenere in considerazione cinque elementi fondamentali che vengono riportati nella sceneggiatura, ovvero le cosiddette 5 W del giornalismo:

- Who: definire i personaggi con le relative caratteristiche è il primo passo fondamentale nella scrittura di una storia.
- What: occorre delineare una trama generale di cosa si intende rappresentare.
- When: bisogna specificare le ambientazioni in cui si svolge la storia.
- Where: è necessario contestualizzare nel tempo gli eventi narrati per rendere maggiormente comprensibile la storia.
- Why: occorre spiegare e motivare quali sono i fattori che determinano la successione degli eventi nella narrazione.

A differenza del soggetto, in cui si deve riportare il racconto in sintesi, la sceneggiatura richiede la descrizione dettagliata, al tempo presente, di ciascun evento che si intende rappresentare, tra cui i movimenti, le gestualità ed i dialoghi tra i personaggi.

La suddivisione in scene rappresenta un metodo efficace che permette di separare gli eventi della storia sia dal punto di vista temporale che spaziale, consentendo maggiore chiarezza espositiva e una miglior comprensione da parte del lettore. La scena viene definita come l'unità narrativa della sceneggiatura, e ciascuna di essa contiene un unico riferimento spazio-tempo. Pertanto, il passaggio da una scena all'altra corrisponde al cambiamento dei riferimenti temporali e dei contesti in cui avvengono gli eventi della narrazione; ogni scena viene inizialmente descritta con i termini «Int.»per far riferimento agli eventi che hanno luogo in spazi chiusi, *interni*, o «Est.»con richiamo agli spazi esterni.
Anche in questo caso, come nel soggetto, occorre descrivere le azioni per immagini, evitando di indicare concetti astratti ma rappresentandoli sotto forma di spiegazioni dettagliate, e limitando l'uso di descrizioni degli stati d'animo dei personaggi.

L'arte di esprimersi attraverso una storia, meglio conosciuta come storytelling, si può considerare come una vera e propria tecnica di comunicazione che nasce dalla necessità di trasmettere emozioni creando così un senso di connessione con i destinatari di riferimento, per trasportarli per un momento in un contesto nuovo definito dalla storia stessa.
Sono diverse le interpretazioni del termine "narrazione", di cui lo scrittore Alan Palmer ne riporta alcune in [10]. Ad esempio, la narrazione viene considerata da Prince come "un discorso che rappresenta uno o più eventi", oppure Jose Landa e Susana Onega la definiscono come "la rappresentazione semiotica di una serie di eventi, significativamente connessi in modo temporale e causale". Mieke Bal definisce invece un evento come "il passaggio da uno stato ad un altro, causato o vissuto dagli attori". Bal inoltre si sofferma su quest'ultimo importante aspetto che

riguarda gli eventi, considerandoli significativi solo se all'interno di una narrazione sono presenti degli attori che vivono tali eventi.
Ma tale concezione che attribuisce all'evento un ruolo significativo all'interno di una narrazione, viene completamente ribaltato dalla scrittrice Monika Fludernik nel 1996, la quale sostiene che il principale elemento di una narrazione non è l'evento ma quella che lei chiama *experientiality*, ossia un'esperienza soggettiva, di coscienza, di sé, quella che Palmer definisce come *fictional minds*. A tal proposito, Palmer individua dodici universi narrativi relativi alle fictional minds, le quali caratterizzano quasi tutti i romanzi; tra questi vi sono le emozioni.

Come si è precedentemente affermato, la volontà di comunicare attraverso una storia deriva principalmente dal desiderio di suscitare emozioni negli interlocutori. Le emozioni infatti sono considerate un aspetto fondamentale per la creazione di un racconto; senza di esse, non si potrebbe più definire racconto ma semplicemente un elenco di informazioni.
Palmer definisce le emozioni come i "motori teleologici della narrazione", "guidano le narrazioni". Gli psicologi Singer e Kolligian affermano invece che "i nostri modi di conoscere il mondo sono intrinsecamente legati ai nostri modi di percepire le sensazioni", mentre lo scienziato cognitivo Antonio Damasio ritiene che "coscienza ed emozione sono inseparabili". Lo studio degli psicologi riguardo il legame tra emozioni e cognizione ha portato a stabilire che tali elementi sono alla base dell'analisi della natura teleologica della narrazione.
Alcuni psicologi ritengono invece che l'emozione sia un concetto discorsivo.

> "Le descrizioni delle emozioni sono atti vocali performativi che si verificano all'interno di giochi linguistici complessi e sono sempre incorporati in contesti sociali specifici."[10]

Infatti è possibile rappresentare l'emozione che un personaggio prova, attraverso una descrizione delle motivazioni per cui tale personaggio compie determinate azioni. All'interno di una narrazione, le emozioni esistono quindi solo come descrizioni in un discorso di finzione.
Tenendo in considerazione le emozioni che solitamente provano i destinatari di una narrazione, gli autori tendono ad introdurre nei loro racconti degli episodi che suscitano nel lettore proprio tali emozioni ricorrenti. Alcuni autori però tendono ad omettere tali emozioni, provocando nel lettore una sensazione diversa da ciò che si aspetterebbe; l'esempio più noto lo si ritrova nel romanzo *L'Etranger*, dello scrittore francese Albert Camus, in cui egli reagisce con indifferenza alla morte della madre.

Un ulteriore aspetto da non sottovalutare per riuscire a catturare l'attenzione dello spettatore è l'introduzione di dettagli ed elementi specifici all'interno della storia. Tali particolari, apparentemente irrilevanti, permettono di caratterizzare la narrazione mostrandola più completa, nonché più coinvolgente.

Un particolare modello che viene in aiuto nella creazione di una storia consiste nella tavola periodica dello storytelling in figura 2.1, progettata dal designer James Harris[11].

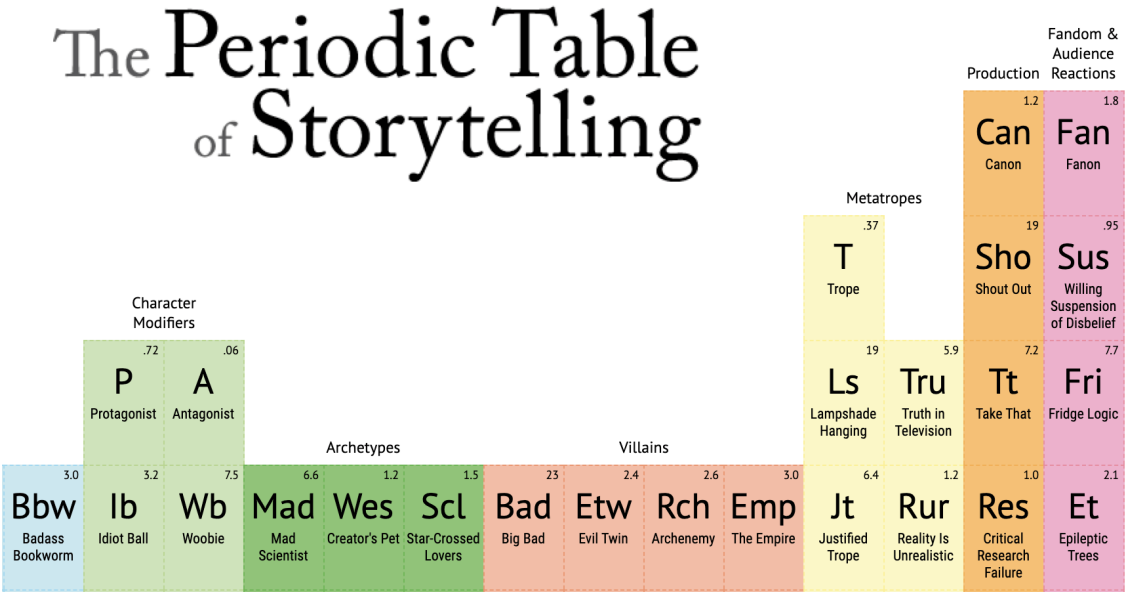


Figura 2.1: Parte della tavola periodica della narrazione, James Harris, 2014. Fonte: [11].

Si tratta di una struttura interattiva organizzata come la tavola periodica degli elementi del chimico Mendeleev, ma in sostituzione dei gruppi di metalli alcalini, dei gas nobili e delle restanti sezioni, vi sono undici gruppi narrativi. Tali categorie, partendo da sinistra verso destra, sono rappresentate da:

- Struttura: contiene i diversi modelli di organizzazione degli elementi di una storia, ad esempio la struttura a tre atti o il triangolo amoroso.
- Ambientazioni, leggi, trame: rappresentano le tipologie di ambienti, le leggi che li governano ed i modelli di trame; tra questi vi è ad esempio Cal, ossia la chiamata all'azione.
- Modificatori della storia: rappresentano i principi che determinato una svolta nella storia, ad esempio il *Jumping the shark*, Jts, ossia il momento della narrazione in cui delle serie consolidate di lunga durata cambiano in maniera significativa.
- Dispositivi: sono gli elementi che forniscono curiosità e aspettative, come Pet, *Pet the dog*. Questo termine, coniato dai sceneggiatori, indica lo stereotipo del personaggio folle che accarezza un cane mostrandosi in pubblico, colui che si accorge che in fondo non è una persona malvagia.

- Eroi: rappresentano i principali tipi di protagonisti della storia, come il Supereroe che ha dei poteri o abilità rispetto alle persone comuni.
- Modificatori dei personaggi: elementi o eventi che portano ad un cambiamento interiore dei personaggi, come Ib, *Idiot Ball*, ossia un momento in cui un personaggio che solitamente è competente diventa improvvisamente incompetente dando vita ad una nuova breve trama.
- Archetipi: rappresentano il carattere dei personaggi in base al ruolo che hanno nella storia; ad esempio il tropo Gia, il Gigante gentile, dall'aspetto spaventoso e dallo sguardo furente, ma dal cuore d'oro.
- Cattivi: rappresentano le tipologie di antagonisti e possono essere personaggi o elementi come il Virus, il quale trasforma le persone in se stesse o in entità sottomesse a se stesse; una trasformazione che può essere mentale o fisica.
- Metatropi: si tratta dei diversi modi con cui si può descrivere la trama della storia; Rur, *Reality is unrealistic*, quando eventi reali vengono raccontati in maniera esagerata al limite dell'invenzione, alcune persone percepiscono la narrazione immaginata più realistica di quanto sia un racconto puramente reale.
- Produzione: riguardano le scelte che l'autore compie nella stesura del racconto. Tra di queste vi è Dh, *Development Hell*, il quale si riferisce allo stato in cui un progetto creativo che è stato annunciato resta bloccato nella fase di preparazione per anni.
- Fandom e reazioni del pubblico: si riferisce alle comunità di appassionati e alle reazioni che questi esprimono, come ad esempio Sqi, *Squick*, un'abbreviazione di squeamish che rappresenta un'espressione di disgusto.

Ogni gruppo presenta un colore diverso a seconda della tipologia di elementi di cui è costituito. In tale contesto gli elementi di ogni gruppo sono definiti *tropi*, ossia figure narrative fittizie che possono essere considerate degli stereotipi. Inoltre ciascun tropo è cliccabile e rimanda alle pagine wiki di Harris su TV Tropes in cui vengono descritti in dettaglio numerosi esempi di storie riguardanti tali figure narrative, come il primo tropo del secondo gruppo identificato dalla sigla Ae che rimanda ad Esopo e rappresenta la morale della favola.

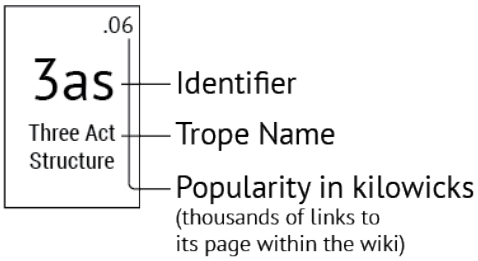


Figura 2.2: Struttura di un tropo. Fonte: [11].

In particolare, per ogni tropo sono riportati i seguenti tre elementi caratterizzanti, visibili in figura 2.2: la sigla, il nome e l'indice di popolarità, dato dalle migliaia di link che rimandano alla pagina wiki del tropo.

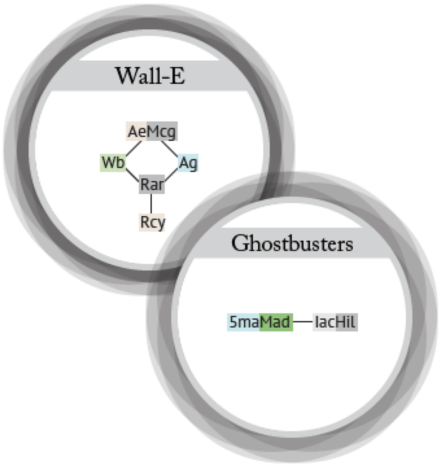


Figura 2.3: Esempi di molecole narrative. Fonte: [11].

Attraverso la combinazione dei tropi contenuti nella tavola narrativa è quindi possibile costruire o scomporre degli intrecci narrativi, quelli che Harris definisce *molecole narrative*, di cui ne riporta alcuni esempi, visibili in figura 2.3. Tali molecole rappresentano le strutture narrative di alcuni celebri film come *Wall-E*, *Star Wars*, *Death Note* ecc. È infatti possibile partire dal selezionare ciascun tropo e creare una storia oppure scomporre una narrazione individuando i singoli tropi di cui è composta.

L'undicesimo tropo appartenente al secondo gruppo della tavola periodica è rappresentato da uno dei più noti modelli narrativi, *Il viaggio dell'eroe*, considerato uno tra i framework concettuali alla base di ogni buon storytelling.

2.2.1 Il viaggio dell'eroe

Nella fase di scrittura di una storia è fondamentale porre attenzione alla graduale evoluzione del protagonista, o eroe, alle situazioni che gli si presentano e al modo in cui egli affronta tali fatti durante il proprio percorso narrativo; si parla quindi del cosiddetto *hero's journey*, ovvero *il viaggio dell'eroe*. Nel caso del cortometraggio, tale figura dell'eroe si immedesima nel personaggio dell'ape Pinnie.

Nel corso degli anni, molti autori sperimentarono diversi intrecci narrativi per creare una struttura che potesse costituire la base di ogni racconto, tra questi vi è Aristotele che teorizzò la forma in tre atti secondo la quale ogni dramma ha un inizio, una parte centrale e una fine, e le parti devono essere tra loro ben proporzionate. Cicerone invece si cimentò nello studio dell'oratoria e della retorica. Mentre lo sceneggiatore statunitense Sydney Alvin Field definì la struttura narrativa articolata in tre atti sulla base del "dramma ben fatto", un modello narrativo del 1800 di Eugène Scribe, in cui egli definì il concetto secondo cui il completo ritorno all'ordine garantisce che tutti gli eventi e situazioni vengano risolti.

Generalmente, nella sceneggiatura di un film di 120 pagine, corrispondente a 120 minuti, il primo e terzo atto sono della durata di 30 pagine, mentre il secondo atto

ha una lunghezza di 60 pagine. Nello specifico, il primo atto di tale struttura narrativa si apre con la presentazione della situazione iniziale in cui vengono introdotti i personaggi, tra cui il protagonista.
Successivamente il protagonista si trova ad un punto di svolta in cui gli si presenta un obiettivo da raggiungere. Generalmente circa metà della narrazione si concentra sulla rappresentazione della lotta di questo personaggio per raggiungere il proprio obiettivo.
Il secondo atto è il confronto; Field accenna anche al punto centrale, un punto di svolta più debole che prevede un ribaltamento apparente della fortuna del protagonista.
Il terzo e ultimo atto è definito dal conflitto vissuto dal protagonista, il quale affronterà la resa dei conti, raggiungendo o meno l'obiettivo finale.

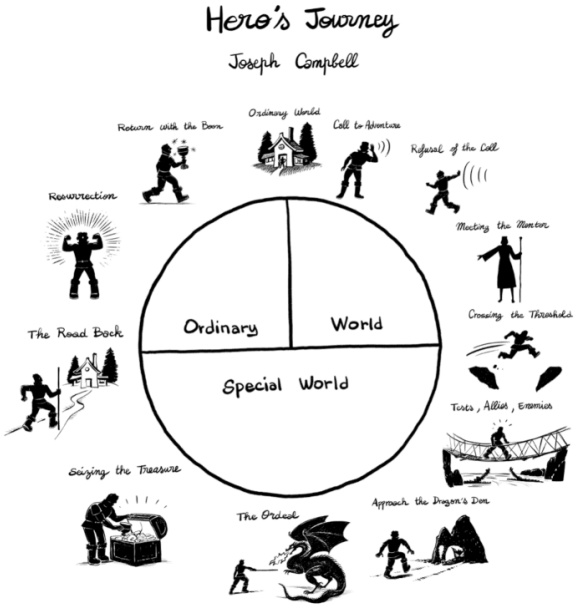


Figura 2.4: Hero's Journey, Park & Company Marketing Communications. Fonte: The Business of Story.

Christopher Vogler attualmente è uno sceneggiatore ad Hollywood, insegna presso l'UCLA e ha preso parte anche alla realizzazione di film quali *La Bella e la Bestia* e la *Sirenetta* prodotti dalla Disney. Negli anni '80 egli analizzò diverse opere e pensieri di celebri autori, tra cui il saggio *Il Ramo d'oro* scritto dall'antropologo Frazer, le fiabe russe di Vladimir Propp e la teoria degli archetipi di Jung. Vogler si ispirò soprattutto al libro *The Hero with a Thousand Faces* ossia *L'eroe dai mille volti* di Joseph Campbell, suo maestro e così nel 1992 pubblicò *The Writer's Journey: Mythic Structure For Writers*, tradotto in italiano con il titolo *Il viaggio dell'eroe*, una sorta di manuale che identifica i ruoli degli archetipi, sostenendo che tutte le storie siano riconducibili a uno schema narrativo elementare articolato in 12 tappe che identificano tale viaggio.

Primo atto

- *Fase ordinaria*: l'eroe è nel suo mondo ideale, un ambiente completamente opposto a quello straordinario. In questa fase viene presentato il protagonista, il quale è caratterizzato da una mancanza ed è con questo personaggio che lo spettatore si immedesima.

- *Chiamata all'avventura*: al protagonista si presenta un evento che lo spinge ad agire. Il mondo ordinario deve quindi subire un danneggiamento.
- *Rifiuto della chiamata*: inizialmente l'eroe prova paura e tende al rifiuto; un rifiuto che può sbloccarsi solo se il protagonista trova una valida motivazione personale che lo porta ad accettare la sfida.
- *Incontro con il Mentore*: il mentore è un ex-eroe che ha già percorso il viaggio dell'eroe, una figura che conoscendo il mondo straordinario è in grado di allenare e consegnare all'eroe tutto ciò di cui ha bisogno per intraprendere il viaggio. Vogler identifica il mentore con la figura di Chirone, un essere mitologico metà uomo e metà animale che a differenza dei comuni centauri propensi alla violenza era il più saggio e gentile.
- *Attraversamento della soglia*: questo è il primo punto di svolta della storia in cui l'eroe lascia il suo mondo ordinario e deve affrontare un primo ostacolo.

Secondo atto

- *Prove, alleati e nemici*: l'eroe entra nel mondo straordinario e continua ad analizzare i suoi avversari.
- *Approccio alla sfida*: l'eroe si prepara a varcare una soglia dalla quale non può più tornare indietro. Vogler sottolinea come l'eroe debba cercare di capire la logica dei suoi nemici per poterli affrontare e sconfiggere.
- *Sfida*: in questa fase l'eroe si trova ad affrontare la prima sfida e prova un pericolo di morte. Vogler definisce questo momento una crisi in cui il protagonista subirà un cambiamento che lo migliorerà. Non è il momento più drammatico della storia, ma un punto in cui l'eroe affronta l'antagonista, che Vogler chiama Ombra, per vincere le sue paure più grandi: far morire il vecchio Io e trovare un nuovo equilibrio.
- *Ricompensa*: dopo un'esperienza di morte metaforica l'eroe riceve un beneficio. Questa fase è caratterizzata da un momento di tranquillità per il protagonista.
- *Ritorno*: l'eroe deve trovare le motivazione per ritornare nel mondo ordinario e portare ciò che ha guadagnato.

Terzo atto

- *Sacrificio e Resurrezione*: è il momento del climax, ossia il momento più intenso in cui l'eroe si trova a dover affrontare una prova soggettiva, nel mondo ordinario, per dimostrare di essere cambiato e di aver appreso come distinguere

il bene dal male. La prova può essere una scelta, una prova fisica o sia fisica che spirituale.

- *Conclusione*: è il momento in cui l'eroe acquisisce saggezza e mostra la sua nuova identità.

Esistono due modelli che rappresentano la conclusione della storia:

- americano: la storia si conclude con tutti i problemi che vengono risolti;
- europeo: la storia è aperta, non tutti i nodi vengono sciolti, ossia i cambiamenti avvenuti nel protagonista non vengono esaltati.

Vogler sostiene che il viaggio dell'eroe deve essere come un elastico, il ritmo della storia deve quindi essere progressivo e alternato da momenti di tensione e di distensione. Inoltre, riferendosi a Propp, egli evidenzia le figure degli archetipi, che possono svolgere diverse funzioni narrative, influenzando il percorso dell'eroe. L'archetipo, dal greco *archè*, origine e *typos*, modello, indica quindi un modello originario. Tale termine fu introdotto in Grecia dai filosofi come Plotino e Proclo per indicare i principi universali, ossia dei modelli preesistenti in natura. Nella tabella 2.1 vengono enunciati i sette archetipi individuati da Vogler e vengono descritti in base alle due funzioni che ricoprono in letteratura: la funzione drammaturgica riferita alla storia e la funzione psicologica riferita al lettore.
Gli archetipi possono esprimere aspetti positivi o negativi, e possono anche mutare nel corso della storia. Ad esempio, il personaggio che ha il ruolo di mentore, può trasformarsi successivamente in ombra.

La scrittura di una storia non deve però essere condizionata esclusivamente da certi schemi narrativi altrimenti si rischia di ottenere qualcosa di già visto, ripetitivo e forzato, pertanto occorre combinare elementi diversi per creare sempre eventi nuovi, trasformare i personaggi ed evidenziare nuovi punti di vista in un racconto.
Il percorso dell'eroe deve quindi essere modulato in base a ciò che si intende comunicare.

Tabella 2.1: I sette archetipi individuati da Christopher Vogler.

	Funzione drammaturgica	Funzione psicologica
Eroe (*Hero*)	Invita il lettore ad immedesimarsi in lui.	Rappresenta l'Io, un'identità unica che si evolve.
Mentore (*Mentor*)	Insegna, dona, motiva, rappresenta la coscienza che aiuta i personaggi.	Rappresenta il Sé, la sicurezza, la parte più saggia di noi.
Guardiano della Soglia (*Threshold Guardian*)	Mette alla prova l'eroe, lo protegge. Rappresenta un qualcosa da cui si può apprendere e crescere.	Rappresenta i nostri demoni interiori: vizi, dipendenze, disagi emotivi. Può essere anche una figura fisica che diventa un alleato.
Messaggero (*Herald*)	Motiva l'eroe offrendogli una sfida da compiere e avverte che è in arrivo un cambiamento.	Indica una svolta, annuncia quindi la necessità di cambiamento.
Mutaforme (*Shapeshifter*)	Suscita dubbi e crea suspense durante la storia.	Indica la crescita psicologica e la conoscenza dell'altro.
Ombra (*Shadow*)	Rappresenta il rivale, l'antagonista che vuole sconfiggere l'eroe e influenza gli altri archetipi.	Rappresenta i sentimenti repressi, le psicosi, emozioni negative, quegli ostacoli da superare.
Imbroglione (*Trickster*)	Permette un intermezzo comico, porta a cambiamenti anche negativi.	Rappresenta il superamento dell'Io, mette in evidenza l'ipocrisia.

2.3 Character design e sfondi

Il *Character design* consiste nello studio e nella rappresentazione grafica dei personaggi che verranno animati. In questa fase vengono prodotti una serie di disegni di prova di ciascun personaggio per delineare la forma, la dimensione, il colore, la fisionomia e particolari espressioni, le posizioni e le posture, gli eventuali indumenti e oggetti particolari che li caratterizzano. I disegni vengono inizialmente realizzati su carta, e successivamente, per una visione più chiara del risultato vengono digitalizzati attraverso opportuni software adatti a tale compito.
È importante analizzare attentamente le varie caratteristiche, elencate precedentemente, per ottenere un risultato efficace sia dal punto di vista visivo, in quanto il personaggio deve avere un certo appeal, sia al fine dell'animazione del personaggio stesso e quindi alla facilità di utilizzo del disegno all'interno dei software. Infatti se troppo complesso, il personaggio potrebbe creare problemi durante la fase di rigging, nella concatenazione dei layer, descritti nel capitolo 3, oppure in casi in cui si necessita di programmare alcuni movimenti.
La struttura del personaggio, e quindi l'articolazione delle parti del corpo è un aspetto fondamentale per la fase di animazione; se tali passaggi, durante la fase di ideazione del personaggio, vengono eseguiti con un'adeguata accuratezza, sarà possibile eseguire un lavoro di animazione più agevole, nonché un maggior risparmio in termini di tempo e di costi.

Dal punto di vista grafico, la creazione di un personaggio richiede gli elementi descritti di seguito[12]:

- stile: lo stile è il primo aspetto da considerare nella fase di creazione di un personaggio e dipende dal tipo di aspetto che si vuole dare al prodotto di animazione, ossia occorre stabilire il tipo di approccio grafico che si intende seguire: cartone animato tradizionale, puramente illustrativo, fumettistico, fotorealistico ecc. Tale scelta deve essere effettuata considerando anche il tipo di ambientazioni che si vogliono ricreare, in modo da evitare un'eventuale incongruenza tra i personaggi e gli sfondi. È inoltre fondamentale tenere a mente che l'uso di forme e tratti arrotondati e curvilinei darà origine ad uno stile più morbido e dolce; viceversa tratti più rettilinei e spigolosi conferiscono un aspetto più aggressivo e tagliente.

- Personalità: ciò che determina l'aspetto di un personaggio è soprattutto la personalità che gli si intende attribuire. È quindi necessario capire se ad esempio il personaggio sarà amichevole, aggressivo, *dumb*, acuto, rilassato, ansioso, attraente, non particolarmente grazioso ecc. Ci sono alcuni stereotipi che di solito vengono seguiti, come il caso del personaggio principale, l'eroe, il quale normalmente non viene disegnato come una figura poco perspicace, ma al contrario è colui che rispetto agli altri personaggi manifesta un maggior senso

di sicurezza. La forma, nonché la natura del disegno sono quindi elementi fondamentali che determinano ciò che lo spettatore percepirà.

- Atteggiamento: l'atteggiamento è un elemento che aiuta a comunicare allo spettatore la personalità del personaggio. Ciò è realizzabile in base alla posizione in cui viene ritratto il personaggio e quindi al modo in cui è disegnata la corporatura. Ad esempio, se il personaggio deve rappresentare una figura ansiosa e insicura, non bisogna disegnarla come se si trovasse in una situazione di combattimento.

- Proporzioni: le proporzioni nella fisionomia del personaggio sono molto utili per caratterizzare maggiormente la figura. Ad esempio, nel caso in cui si scelga di disegnare il capo di dimensioni notevolmente maggiori rispetto al corpo, ciò suggerisce che il personaggio è intelligente; oppure un personaggio dalle spalle molto larghe, con la mascella quadrata raffigura un personaggio coraggioso e propenso all'azione. Tali tipologie di rappresentazione possono però variare in base ai casi.

- Altezza del capo: l'altezza della testa del personaggio è uno tra gli elementi più importanti nel design di un personaggio, in quanto oltre a rappresentare l'altezza totale della figura e la dimensione della testa in relazione al corpo, definisce quanto il personaggio è realistico o quanto si avvicina maggiormente ad una caricatura. Tipicamente, l'altezza della testa si calcola in base al numero di volte in cui la testa del personaggio è divisa rispetto all'altezza totale del personaggio; ossia l'altezza della testa umana è definita come sette, il che significa che la testa occupa un settimo della dimensione totale del corpo. Nel caso di personaggi dei cartoni animati, la testa può avere una dimensione inferiore a due.

- Silhouette: la sagoma è l'elemento che definisce immediatamente le caratteristiche di un personaggio. Infatti, osservando solamente le silhouette di colore nero rappresentante personaggi animati famosi, è possibile riconoscere all'istante a quale personaggio appartengono. Creare una silhouette che abbia tutte le caratteristiche necessarie per rendere il personaggio particolare è quindi un aspetto fondamentale che permette al proprio personaggio di essere ricordato dallo spettatore in qualsiasi contesto.

- Dettagli: per quanto riguarda soprattutto un lavoro di animazione in 2D, occorre tenere in considerazione l'uso dei dettagli che si intende inserire nel design di un personaggio. Questi infatti, da una parte possono arricchire l'originalità del personaggio ma d'altra parte tendono a complicare il lavoro di produzione. Ad esempio, un personaggio dovrebbe essere ridisegnato più volte nel corso dell'animazione e se presenta numerosi dettagli ciò potrebbe costituire un problema sia dal punto di vista della complessità produttiva, sia riguardo al

tempo di lavorazione richiesto. Pertanto in questo caso, è consigliabile creare un personaggio minimale che abbia pochi dettagli per facilitare la successiva fase di animazione.

Oltre al design fisico di un personaggio, è importante anche l'aspetto psicologico che caratterizza tale personaggio ed è ciò che interessa e coinvolge maggiormente lo spettatore. In [13], Chris Webster afferma:

> "Per una soluzione che sia davvero di successo entrambi gli elementi, buon design abbinato ad una performance ben realizzata, sono necessari. Se il designer non conosce e non crede nei personaggi e l'animatore non riesce a coinvolgersi emotivamente con il suo soggetto, allora nemmeno un pubblico ha alcuna possibilità e il personaggio rimarrà poco coinvolgente."

I movimenti e le espressioni di un personaggio costituiscono l'aspetto fondamentale alla base del processo di animazione. Ciò che dà vita alla sceneggiatura è proprio il modo in cui essa viene rappresentata, nonché le performance che compie ciascun personaggio all'interno degli ambienti.

> "I tuoi personaggi non dovrebbero semplicemente muoversi, illustrando servilmente la sceneggiatura, essi dovrebbero mostrare lo stato d'animo, il temperamento e le emozioni. Essi dovrebbero permetterci di immedesimarci in loro; dovremmo sapere non solo cosa pensano, ma anche cosa provano."

Webster inoltre indica tre considerazioni che occorre seguire per riuscire a creare un personaggio che sia il più possibile coinvolgente:

- conoscere i propri personaggi.
- Empatizzare con i propri personaggi.
- Diventare i propri personaggi.

Tali punti intendono sottolineare l'importanza dello studio dei personaggi e la rappresentazione delle emozioni attraverso tali figure. Ogni personaggio rappresentato deve infatti essere spinto da una motivazione che lo porti a raggiungere un obiettivo. Affinché lo spettatore comprenda e dia senso alle azioni svolte dai personaggi e apprezzi le loro motivazioni, è necessario innanzitutto definire tali obiettivi e motivazioni psicologiche che suscitano certi comportamenti ed emozioni. Inoltre, al fine di creare un'animazione credibile, è necessario che lo spettatore si connetta con i personaggi della storia empatizzando con essi. Ancor prima però occorre che tale connessione si crei tra l'ideatore dei personaggi ed i personaggi stessi; bisogna quindi capire il tipo di carattere ed emozioni associare a tali figure, riconoscersi in esse. In questo modo sarà più facile trasmettere allo spettatore le sensazioni provate dai personaggi ed immedesimarsi in loro.

Siccome il cortometraggio tratta un argomento reale in cui i principali protagonisti sono gli animali, si è inizialmente effettuato uno studio riguardo le principali specie

a rischio estinzione, analizzandone la fisionomia, le caratteristiche che li contraddistinguono, le abitudini comportamentali ed i casi verificatisi che li hanno coinvolti nelle problematiche ambientali.
Gli stessi studi e ricerche sono stati messi in pratica anche per la scelta degli ambienti che definiscono il contesto in cui sono inseriti i personaggi, in modo da renderli veritieri ed evidenziare problematiche reali. D'altra parte, trattandosi di un video di animazione, gli aspetti reali vengono combinati ad elementi di pura fantasia.

Sfondi

Nel caso dell'animazione 2D, prima di iniziare la fase di animazione è fondamentale sia definire quali elementi devono essere animati e quali invece rimangono fissi, sia sapere le posizioni specifiche di tali elementi per ciascuna scena. Anche in questo caso esistono delle regole da seguire in fase di design che vengono riportate di seguito [12]:

- distanza e prospettiva: per caratterizzare al meglio una scena occorre articolarla con primi piani, punti centrali tra il background e il foreground e con viste in lontananza. Utilizzando tali elementi a diverse distanze e prospettive, si potrà ottenere una sensazione di profondità alla vista dell'inquadratura, sia per quanto riguarda gli ambienti interni sia per quelli esterni come i paesaggi.
- Focus dell'attenzione: generalmente l'animazione si sviluppa al centro di una scena. Di conseguenza, quando si progetta il layout degli sfondi occorre che le linee, di solito quelle prospettiche, e gli oggetti presenti confluiscano nel punto in cui si desidera focalizzare l'attenzione, e quindi nel punto in cui si svolgeranno le azioni principali.
- Azioni di silhouette: per ottenere un'animazione efficace occorre che questa sia incorniciata, ossia venga messa in risalto rispetto al resto degli elementi presenti nella scena, altrimenti si otterrà un impatto minore sul personaggio in questione in quanto le azioni da egli svolte passeranno in secondo piano.

Dopo aver definito le linee guida che caratterizzano i personaggi e gli sfondi, sono stati realizzati dei disegni a matita che ritraggono ciascun personaggio da diverse angolazioni a seconda delle posizioni che occupano nelle scene del cortometraggio. Di seguito vengono riportati alcuni degli sketch realizzati con le relative descrizioni; in alcuni casi sono riportati riferimenti relativi ad episodi realmente accaduti che hanno coinvolto le specie animali trattate e che hanno costituito le reference di base.

Pinnie

Pinnie è la protagonista della storia e rappresentante dell'ecosistema delle api, colei che si identifica nella figura dell'osservatore e manifesta la volontà di iniziare un cambiamento.
Nel corso della storia, l'ape Pinnie rappresenta una guida per l'utente, la quale, verso la fine della narrazione, cederà il suo ruolo da protagonista all'utente dell'interazione, il quale avrà il compito di portare a termine una missione per poter sbloccare la situazione di sciopero indetta dagli animali apportando così dei cambiamenti positivi.
Pinnie è dalla corporatura esile ed una delle caratteristiche che la contraddistinguono dalle altre specie animali è la presenza di un paio di antenne con delle sporgenze a forma di cubo, grazie alle quali è in grado di captare i vari pericoli per poi visionarli sui dispositivi dalle fattezze cubiche. Nella figura 2.5 sono ritratte tre angolazioni di Pinnie.

(a) Pinnie di profilo **(b)** Pinnie di 3/4 **(c)** Pinnie frontale

Figura 2.5: Illustrazioni di Pinnie

Beekeeper e Fisherman

L'apicoltore Beekeeper ed il pescatore Fisherman rappresentano il genere umano, l'antagonista del mondo animale. Essi simboleggiano l'*Io* come singolo individuo che agisce considerando che le proprie azioni non abbiano un impatto significativo sul nostro pianeta, figura 2.6.

(a) Beekeeper di spalle

(b) Fisherman di 3/4

Figura 2.6: Illustrazioni di Beekeeper e Fisherman

Fish

Fish rappresenta la specie dei pesci rari che vivono solamente in acque cristalline, ma la contaminazione e la distruzione anche di questi habitat fa sì che questi pesci siano prossimi all'estinzione.
Presenta una fisionomia caratterizzata da una lunga coda grazie alla quale può difendersi dai pericoli, figura 2.7.

Figura 2.7:
Fish di profilo

Octopus

Figura 2.8:
Octopus frontale

Il polipo Octopus indossa un giubbotto catarifrangente e vende giornali per le strade di Fishadelphia; tiene sempre lo sguardo imbronciato, figura 2.8.
Rappresenta la figura che esprime rassegnazione, colui che si è adattato alle condizioni presentatesi e svolge unicamente e ripetutamente il suo lavoro solamente per sopravvivere, annullando ogni speranza che qualcosa possa cambiare in positivo.

Mrs. Cuttlefish ed i tre pesci

I tre pesci seduti nell'aula dell'asilo rappresentano le nuove generazioni che hanno bisogno di essere istruite sin da bambini per potersi difendere, nonché capire quale

sia il modo migliore per apportare dei cambiamenti.
La seppia Mrs. Cuttlefish rappresenta invece la figura dell'istruttrice. Tali personaggi sono visibili in figura 2.9.

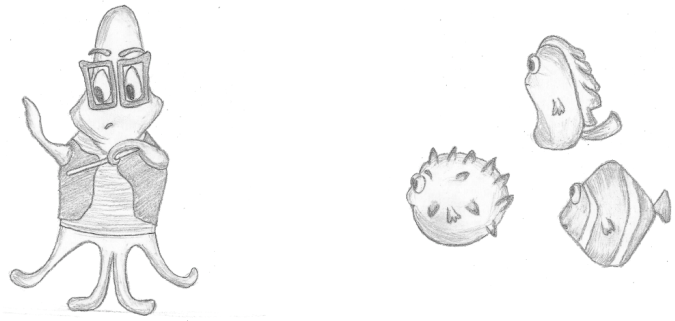


Figura 2.9: Mrs. Cuttlefish ed i tre pesci

Turtle

La tartaruga marina Turtle rappresenta la fragilità e l'impotenza di poter reagire ad una situazione in cui ne è la vittima. È colei che, insieme ad altre specie animali, è costretta a subire passivamente le conseguenze provenienti dall'esterno, figura 2.10. In alcuni casi è possibile che tali animali riescano a salvarsi, altre volte non si ha più tempo per rimediare, come nel caso di Cetacean.

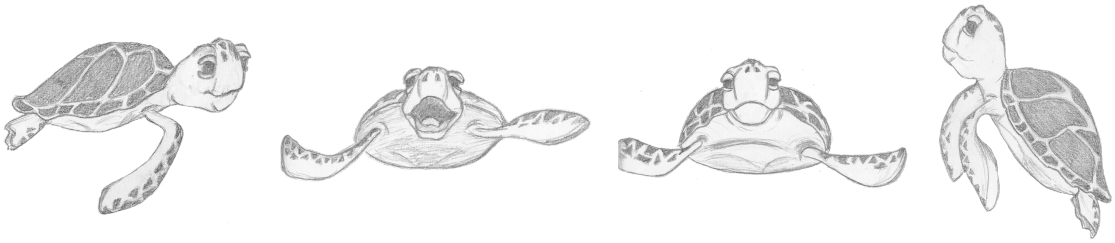


Figura 2.10: Turtle ritratta da diverse angolazioni

Cetacean

Cetacean è il capodoglio che simboleggia le specie animali che ogni giorno, non riuscendo ad arginare i danni causati dall'uomo, si estinguono in condizioni atroci. Per la rappresentazione del capodoglio spiaggiato è stata usata una reference di uno dei numerosi episodi realmente accaduti; nello specifico tale episodio si è verificato a dicembre del 2019 sulla spiaggia di Seilebost, sull'isola di Harris nel Regno Unito.

Si tratta di un capodoglio non ancora adulto morto a causa di una massa di rifiuti di circa 100 kg trovata nello stomaco, figura 2.11.

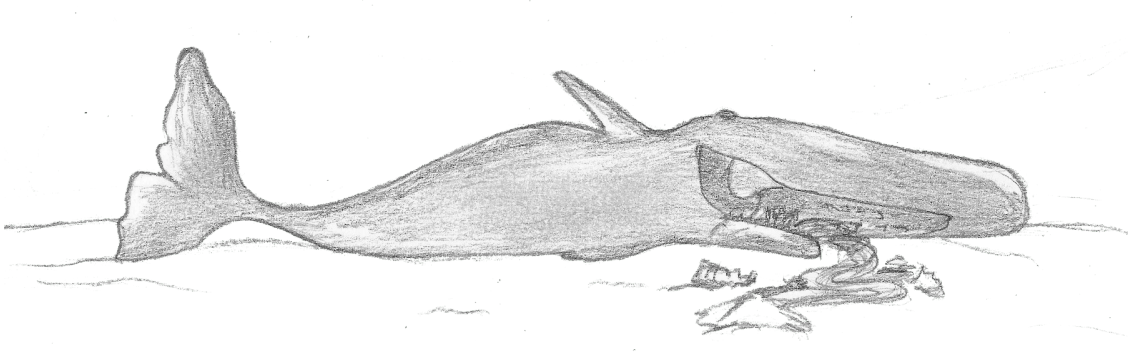


Figura 2.11: Cetacean frontale

Gull

Il gabbiano Gull rappresenta la figura che esprime ingenuità, colui che insieme ad altre specie animali non riesce a distinguere ciò che realmente vengono considerati dei pericoli causati dall'uomo.
Nella storia Gull, portato dall'istinto di recuperare cibo per nutrire sé ed i suoi piccoli, confonde un mozzicone di sigaretta per un alimento di cui è solito cibarsi, figura 2.12.
Anche in questo caso si è preso ispirazione da vari episodi che accadono sempre più frequentemente in cui si vedono realmente pulcini di gabbiano che tentano di mangiare dei mozziconi di sigaretta portati dalle loro madri.

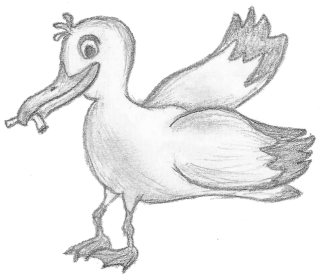


Figura 2.12: Gull di profilo

Bear

Figura 2.13: Bear frontale

Bear è un orso polare che simboleggia colui che cerca di adattarsi ai cambiamenti climatici, evitando il più possibile i vari rischi a cui va incontro, figura 2.13.

L'albero delle api

L'albero delle api è l'ambientazione che rappresenta il centro operativo in cui vivono e lavorano le api. Si tratta di una casa intagliata in un albero in cui una delle sue sporgenze presenta un cubo dotato di luci che svolgono il ruolo di segnalazione di un pericolo ambientale in cui vengono coinvolti gli animali a livello globale, figura 2.14. Tale cubo infatti lampeggia di rosso quando si è appena verificato un caso, richiamando così l'attenzione delle api, in particolare di Regina che tiene sotto controllo tali avvenimenti.

Figura 2.14: L'albero delle api

Successivamente questi disegni sono stati riprodotti digitalmente sul software Moho e in seguito è stata eseguita la fase di colorazione sulla base dei principi studiati che permettono di ottenere un risultato piacevole alla vista, in funzione dei contesti in cui i disegni devono essere inseriti. Tali aspetti verranno trattati nel capitolo successivo.

2.4 Storyboard e animatic

Dopo aver ideato e scritto la trama della storia, il passaggio successivo consiste nella rappresentazione della narrazione, per cui lo storyboard ne è la massima espressione.

Con il termine storyboard, letteralmente tavola della storia, si intende uno strumento cartaceo o digitale per creare una rappresentazione grafica e semplificata delle sequenze di un contenuto che si intende descrivere. Si può definire come una sintesi visuale, solitamente disegnata, di un lavoro più articolato che verrà sviluppato in seguito.
Viene anche utilizzato come metodo alternativo alla sceneggiatura; infatti molti animatori che non dispongono di una buona formazione nella scrittura, preferiscono utilizzare lo storyboard come script per la propria storia.
In concreto, tale strumento deve rappresentare la storia nella sua interezza, consiste quindi in una sequenza di riquadri, inquadrature, all'interno dei quali vengono disegnati gli eventi chiave della storia, i colpi di scena e gli spostamenti dell'ambientazione. Accanto o al di sotto dei riquadri vengono riportate anche delle didascalie che descrivono a parole l'azione raffigurata, i movimenti di camera, i dialoghi e gli effetti sonori. Ciascuna inquadratura è identificata dal nome e dal numero della scena in modo da avere un ordine più chiaro e facilitare il lavoro di produzione.
Non esiste un unico modo di delineare uno storyboard, infatti dipende dal contesto e dal tipo di prodotto che si vuole realizzare, ad esempio può contenere solo le didascalie senza i disegni; nel caso della produzione dei cortometraggi, e video in generale, è utile riportare frecce che suggeriscono le azioni dei personaggi, gli effetti di transizione fra le inquadrature e i movimenti di camera.
La lunghezza dello storyboard dipende dalle azioni previste per l'animazione, mentre i disegni illustrati possono essere semplici schizzi o immagini molto accurate, ma in entrambi i casi non coincidono con il risultato reale del prodotto finito.

Inoltre la fase di ricerca di reference costituisce un valido aiuto per la realizzazione dello storyboard in quanto consente di avere delle basi da cui prendere ispirazione, sia per quanto riguarda i personaggi ma soprattutto per gli sfondi nel caso in cui debbano essere verosimili; tali reference possono essere immagini, video reperibili sui libri, sul web oppure materiale prodotto direttamente in sopralluoghi, nel caso di luoghi specifici, o semplicemente osservando ciò che incontriamo quotidianamente. Queste ultime fonti reperibili in prima persona, in contesti in cui ciò è possibile, sono quelle maggiormente affidabili e consigliate.
Durante la fase di realizzazione dello storyboard si decide la disposizione dei personaggi all'interno degli ambienti e le azioni che essi compiono per ogni inquadratura. Pertanto, avendo un'idea generale dell'intera storia e non essendo definitiva, è possibile apportare delle modifiche nei punti in cui il ritmo narrativo risulta poco fluido o ci si accorge della presenza di incongruenze narrative.

Prima di effettuare la realizzazione dei disegni necessari all'articolazione dello storyboard, si è stilato un elenco contenente ogni scena richiesta dalla sceneggiatura, successivamente ciascuna scena è stata suddivisa in inquadrature. Queste ultime sono state numerate in modo da avere una stima iniziale della quantità di disegni necessari, nonché un riferimento tempistico del lavoro di produzione. Tali inquadrature sono state descritte tenendo in considerazione i personaggi previsti, le azioni che essi avrebbero dovuto svolgere ed il contesto in cui sono stati inseriti.
In figura 2.4, sono riportate alcune miniature che compongono lo storyboard del cortometraggio.

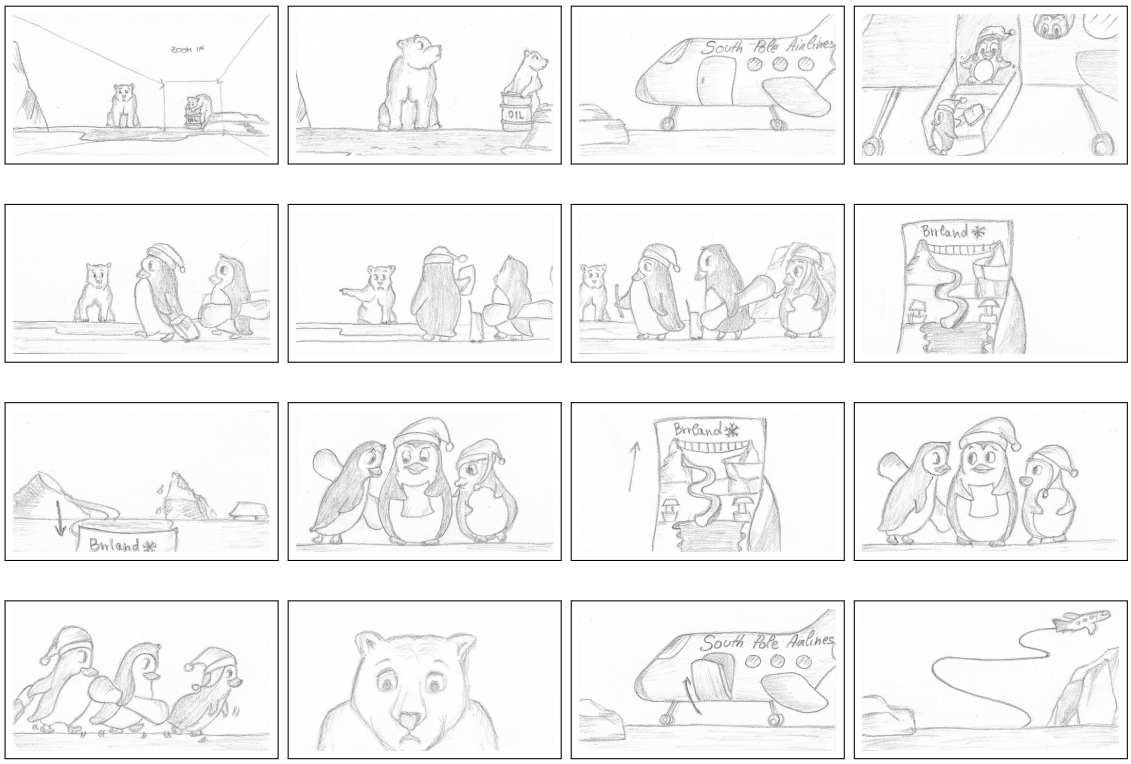


Figura 2.15: Alcuni disegni dello storyboard del cortometraggio.

Animatic

L'animatic detto anche story reel, consiste in un montaggio video primordiale delle inquadrature rappresentate nello storyboard con l'aggiunta di una traccia audio per una visione più precisa delle scene da rappresentare. È una pratica utile in quanto permette di individuare approssimativamente la durata parziale delle singole animazioni e la durata completa del video. Di conseguenza, consente di compiere una scelta logistica sulla necessità di mantenere o di rimuovere scene superflue che non porterebbero alcun interesse nella continuità narrativa della storia. Infatti per

ottenere un buon prodotto di animazione occorre che ogni scena ideata e le azioni più significative siano inserite in maniera chiara ed efficace all'interno della messa in scena e del layout.

Il *timing* è quindi un aspetto fondamentale da tenere in considerazione nel processo di animazione. Se una sequenza che richiede determinati movimenti rapidi per suscitare pericolo, paura o divertimento viene animata in maniera troppo lenta, l'animazione risulterà poco fluida, distogliendo l'attenzione dello spettatore; d'altra parte una dinamica troppo veloce per sequenze che richiedono suspense o tristezza, creerà un senso di disorientamento nello spettatore.
Per comprendere come dosare al meglio questi elementi occorre avere una consapevolezza di come la mente dello spettatore reagisce alla visione di una storia, oltre al fatto che differenti tipi di animazione richiedono approcci di tempo diversi.
A differenza di un film basato su riprese dal vivo, un prodotto di animazione richiede una maggiore attenzione nella fase di ideazione delle scene, in quanto, una volta completata la fase di disegno, sarà più difficile apportare delle modifiche. Pertanto l'obiettivo del produttore è rendere il lavoro di animazione efficiente sia in termini economici che tempistici. L'animatic è quindi uno strumento che permette di inserire diversi elementi guida fondamentali nella fase di produzione: numero di inquadrature, azioni che si svolgono all'interno delle inquadrature rappresentandole attraverso frecce o riquadri per indicare i movimenti di camera o effetti come le transizioni, e la durata della singola inquadratura.
Il concetto di timing come elemento da adoperare per ottenere un'animazione efficace verrà trattato nel paragrafo dedicato ai 12 principi dell'animazione.

Le immagini in figura 2.16 ritraggono alcune inquadrature appartenenti ad una scena che si è deciso di rimuovere in quanto è stata considerata superflua e avrebbe rallentato il ritmo richiesto dalla narrazione, nonché non sarebbe stata più adatta al tipo di interazione successivamente studiato per il progetto. Nello specifico, tale scena ritrae il momento in cui la protagonista Pinnie, dopo aver lasciato il centro operativo, si trova sulla spiaggia di un'isola. Attendendo ordini da Regina riguardo lo sciopero degli animali, Pinnie è intenta a rilassarsi seduta sopra una noce di cocco, mentre due formiche le portano del miele da sorseggiare. Una di loro, in piedi sopra il carapace di un granchio, dopo aver aperto la noce di cocco con un cavatappi, inserisce nel foro una cannuccia di plastica. Pinnie si accorge immediatamente del fatto e tenta di bloccare la formica dall'inserire tale cannuccia. La formica, rammaricata dell'accaduto, estrae subito la cannuccia e la porta via con sé.
Trattandosi di un cortometraggio interattivo, a questo punto della narrazione era previsto un intervento da parte dell'utente, mediante il cubo fisico, nel decidere in quale tra i contenitori della raccolta differenziata, visibili tramite una grafica, sarebbe stato corretto gettare la cannuccia. In base alla scelta effettuata ci sarebbe stata una continuazione diversa della storia. Nel caso in cui la scelta da parte dell'utente fosse stata quella corretta, la narrazione sarebbe proseguita con Pinnie

che sorseggia il cocktail di miele da una cannuccia in bambù, ossia un materiale biodegradabile. Ad un tratto Pinnie riceve un messaggio da Regina in cui le si dice di iniziare il suo viaggio verso i diversi habitat in cui si trovano gli animali maggiormente a rischio di estinzione per raccogliere le adesione allo sciopero indetto dall'ecosistema delle api. Da questo punto la storia riprende, con Pinnie che prosegue nel primo ecosistema.

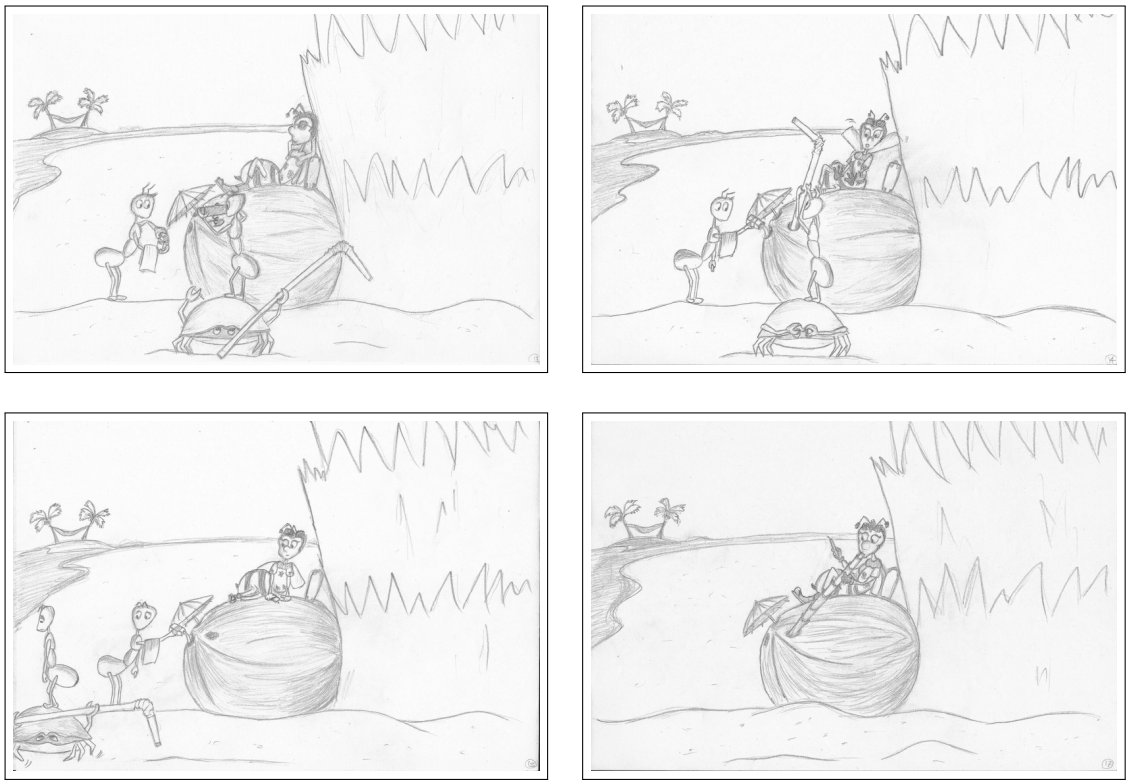


Figura 2.16: Alcune inquadrature di una scena tagliata dal cortometraggio

Capitolo 3

Le fasi di lavoro: Produzione

— Elisa Musolino

Nella fase di produzione verranno trattati i passaggi impiegati nella realizzazione effettiva del cortometraggio di animazione, dal disegno al risultato finale. Saranno inoltre approfonditi particolari aspetti teorici, tra cui i principi base dell'animazione, e saranno affiancati da immagini illustrative tratte dal cortometraggio.
La figura 3.1 mostra il workflow seguito per questa parte di progetto.

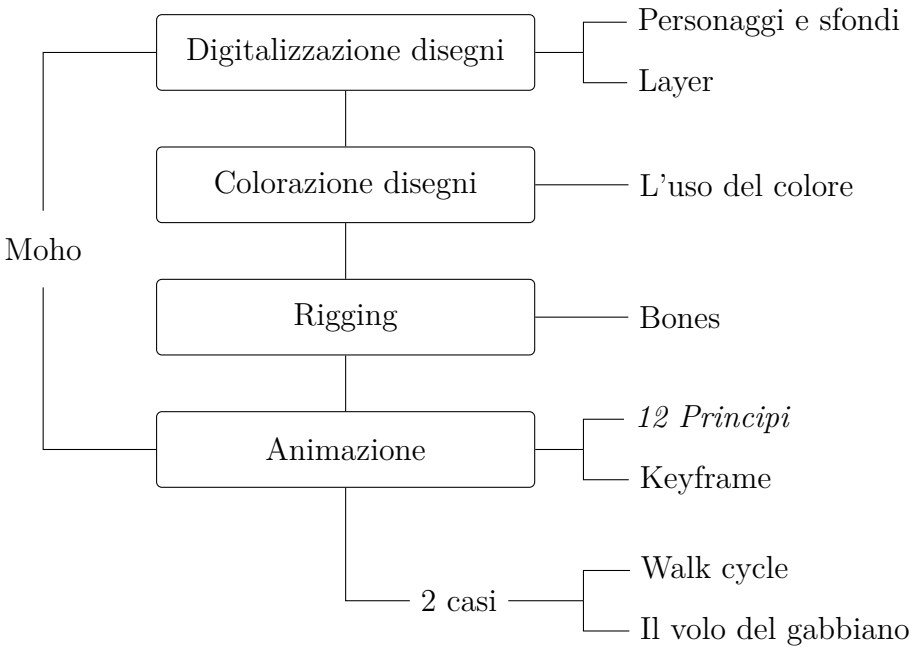


Figura 3.1: Workflow del cortometraggio

3.1 Moho

Moho è un software di animazione basato sulla grafica vettoriale che viene adoperato per produzioni di animazione 2D. È stato sviluppato da Mike Clifton presso la compagnia di sviluppatori software Lost Marble nel gennaio del 1998 e nel corso degli anni è stato ampliato con specifiche funzioni rilasciando numerose versioni del programma [1]. Anche il nome ha subito delle variazioni da una versione all'altra; nel 2007, quando passò alla compagnia Smith Micro Software, prese il nome di Anime Studio. Il 9 giugno 2009 fu rilasciata la versione Anime Studio Pro 6 che presenta diverse caratteristiche quali motion tracking, lip-sync, scatter brush, auto shading, morph targets, sequencer, scripting con LUA 5.1 e il supporto di video in HD. L'ultima versione sotto il nome di Anime Studio Pro 11 risale invece al 3 giugno del 2015 e offre una moltitudine di nuove funzionalità: modalità frame by frame, posizionamento dei layer, disposizione delle *shape* animate, targets delle ossa, imparentamento delle ossa, miglioramento dei tool e pennelli, miglioramento del supporto dei file di Photoshop, bone flipping, miglioramento del smart bone, raggruppamento dei layer, normalizzazione del layer scale, perfezionamento della timeline, miglioramento degli stili, esportazione batch ecc.
Il 9 agosto 2016, Smith Micro Software cambiò nuovamente il nome del software ritornando a Moho, rilasciando recentemente la dodicesima versione, per evidenziare l'enorme potenzialità che raggiunse rispetto ad un programma per anime.

La scelta del software da utilizzare per la realizzazione del cortometraggio oggetto di tesi è ricaduta proprio su quest'ultima versione, Moho 12, in quanto si è voluto sperimentare l'utilizzo di un nuovo programma, studiando e valutando l'efficacia delle molteplici funzionalità offerte.
In particolare, tale versione dispone di nuovi tool ed alcune caratteristiche migliorate rispetto alle versioni precedenti, che si sono rivelate particolarmente utili nella produzione del cortometraggio. Di seguito vengono descritte brevemente alcune di queste funzionalità tipiche di Moho 12, sulla base di quanto riportato nel manuale ufficiale [2]:

- Finestra Switch Selection: i livelli di switch possono essere selezionati più facilmente mediante l'introduzione di una nuova finestra separata. Tale funzione consiste nel visualizzare in una finestra l'anteprima di tre diverse pose di un oggetto posto ciascuno su un layer; tramite uno slider predisposto è possibile passare da una posa all'altra e in base a quella scelta verranno creati automaticamente dei keyframe nella timeline per memorizzare tale posa. È utile in casi in cui ad esempio si vuole animare un personaggio in cui solamente alcune parti del corpo devono cambiare nel corso del tempo.
- Auto-Freeze Keys: permette di aggiungere un keyframe per ciascun osso di un personaggio quando si aggiusta un movimento. Tale funzione consiste nel

memorizzare delle pose del personaggio in determinati frame della timeline ed è utile quando si intende replicare una determinata posizione successivamente ad altri keyframes creati.

- Pin Bones: lo strumento Bone permette di aggiungere Pin Bones, ovvero ossa con lunghezza nulla. Tale tipologia di osso può essere utilizzata ad esempio quando si intende creare un punto di rotazione come nel caso di un braccio: il pin può essere posto a livello del gomito in modo da consentire il piegamento del braccio.

Per quando riguarda la versione Pro, quest'ultima è stata integrata con ulteriori strumenti ancor più efficaci che hanno permesso una semplificazione del workflow di produzione:

- Bezier handlers per i vettori: Moho ora supporta di quelle che possono essere tradotte letteralmente come 'maniglie' di Bezier, tipiche di programmi di grafica vettoriale, quali ad esempio Adobe Illustrator. Tali maniglie sono rappresentate da linee che compaiono in prossimità dei punti che costituiscono il bordo di una shape e grazie a queste è possibile controllare con maggiore facilità le curve vettoriali nei punti in cui il tratto non appare come si desidera.
- Esportazione dei vettori: Moho consente di esportare sequenze di file SVG che possono essere modificate utilizzando software di editing vettoriale, come Adobe Illustrator.
- Modifica di più layer nella timeline: è possibile visualizzare contemporaneamente più layer nella timeline, ognuno eventualmente etichettato con il proprio colore. Ciò consente ad esempio di modificare la posizione di più layer contemporaneamente, ottimizzando i tempi di lavorazione.
- Mute channels: i canali di animazione possono essere disattivati dalla timeline in modo da poter modificare solamente quei canali che si utilizzano al momento. Ad esempio nel caso in cui si sta lavorando su una scena o su di un personaggio che prevede molteplici movimenti di layer o di camera, è possibile disattivare momentaneamente qualsiasi keyframe che non si vuole modificare e concentrarsi unicamente su un elemento.

Rispetto agli altri software in circolazione per l'animazione, come Toon Boom Harmony, Adobe Animate, precedentemente denominato Flash, Toonz, Krita e Blender, Moho è il programma che presenta una maggior potenzialità e versatilità dal punto di vista delle funzioni che riguardano l'animazione.
Nella tabella 3.1 è riportata una panoramica che mette a confronto le principali caratteristiche di alcuni software di animazione, tra cui Moho Pro 12.

Tutte le fasi di produzione che verranno descritte nei paragrafi successivi sono state eseguite esclusivamente su Moho 12. Il funzionamento dei principali strumenti

Tabella 3.1: Principali caratteristiche di alcuni programmi (da sinistra): Harmony, Animate, TVPaint, Toonz, Open Toonz, Clip Studio Paint, Krita, Blender, Moho. L'asterisco indica una funzionalità limitata. Fonte: MultimediArts. Dati aggiornati al 20/07/2019.

Bitmap Painting	✓		✓	✓	✓	✓	✓		✓
Vector Tools	✓	✓	✓	✓	✓	✓	✓*	✓	✓
Rigging	✓	✓*		✓	✓			✓	✓
Point animation to Vectors								✓	✓
Smart Warp for Vectors and bitmap images	✓	✓		✓*	✓*			✓	✓
Other deformation tools	✓	✓	✓	✓	✓	✓	✓	✓	✓
Frame by Frame	✓	✓	✓	✓	✓	✓	✓	✓	✓
3D Workspace	✓	✓		✓	✓			✓	✓
Import PSD File	✓	✓				✓	✓		✓
Camera tools	✓	✓		✓	✓	✓*		✓	✓
Animation Curves Keyframe	✓	✓		✓	✓	✓*	✓*	✓	✓
Visual Effects	✓			✓	✓			✓	✓
Audio Import	✓	✓	✓	✓	✓	✓	✓	✓	✓
Import OBJ or other 3D files	✓	✓				✓		✓	✓

utilizzati verrà illustrato in dettaglio in relazione al tipo di procedimento richiesto nelle diverse fasi della pipeline rappresentata precedentemente.

3.2 Disegno e colorazione

La digitalizzazione dei disegni è stata effettuata mediante l'utilizzo di una tavoletta grafica, uno strumento ormai integrato con qualsiasi software per la grafica e che semplifica notevolmente tale fase di disegno permettendo una maggiore precisione e controllo nei tratti che vengono generati, a differenza dell'uso del classico mouse. In particolare, per tale fase è stato utilizzato il tool Freehand, accennato nel paragrafo precedente, di cui dispone Moho, sfruttando la pressione della penna sulla tavoletta per ottenere tratti con spessore variabile in alcuni punti della figura; successivamente tramite i tool *Transform points* e *Curvature* sono stati corretti i punti in cui il tratto, ossia lo *stroke*, non risultava omogeneo.
Per quanto riguarda la scelta del pennello, si è optato in gran parte per l'uso del brush basilare di colore nero per creare le sagome dei personaggi e degli sfondi, mentre per ottenere particolari tipi di effetti visivi sono stati usati diverse tipologie di pennelli contenenti delle texture, come nel caso della spuma delle onde del mare e le bolle di ossigeno presenti sott'acqua restringendo ed allargando l'ampiezza del tratto nei punti necessari mediante il comando *line width*.

Anche la maggior parte dei contenuti testuali presenti nel cortometraggio sono stati realizzati tramite l'uso del brush semplice, tranne nel caso visibile in figura 3.2 per il quale è stato usato un pennello dalla texture contenente dei rettangoli distanziati, disposti a seconda della lettera, del numero o della figura necessaria per dare forma al testo. La scelta di utilizzare dei pennelli anche per i testi è stata effettuata al fine di avere una maggiore libertà nel posizionamento e nella forma da attribuire alle singole lettere, nonché per mantenere lo stile grafico ed integrare maggiormente le scritte nel contesto rappresentato nel cortometraggio, aspetto che altrimenti sarebbe risultato più limitante e incongruente optando per l'uso di font predefiniti e già esistenti.

Figura 3.2: Testo realizzato mediante un brush dalla texture tratteggiata.

Per quanto riguarda i personaggi, questi sono stati realizzati, a seconda della scena prevista, di pari passo con gli sfondi, scegliendo inoltre di produrre una scena alla volta per avere fin da subito un'anteprima del risultato finale.
Inizialmente è stato realizzato un cosiddetto *model sheet* digitale, ossia un file in cui è stato disegnato ciascun personaggio, definendone le dimensioni, le proporzioni e le varie espressioni che lo caratterizzano in base a quanto ideato nella fase del character design. Ciascun disegno del personaggio è raffigurato da diverse angolazioni: principalmente la vista frontale, il profilo e la vista posteriore. Per semplificare il

lavoro di produzione sono stati realizzati fin dall'inizio i personaggi ritratti nelle pose studiate e rappresentate nello storyboard.
In figura 3.3 vi è una raccolta di alcuni disegni che ritraggono la protagonista Pinnie in diversi contesti previsti dalla storia, nonché diversi elementi che caratterizzano maggiormente il personaggio.

Figura 3.3: Model sheet di Pinnie

Layer

La progettazione della struttura dei personaggi è stata articolata in base al tipo di fisionomia che caratterizza il corpo e al tipo di movimento che si è voluto ricreare, disponendo ciascun elemento di cui è costituito il personaggio su un livello diverso, figura 3.4.
Per gli sfondi si è utilizzato lo stesso approccio considerando invece l'aspetto relativo alla prospettiva, nonché articolando in diversi layer la disposizione degli elementi che costituiscono la scena.
Le immagini quindi sono state create suddividendo la scena in diversi elementi e assegnando a ciascuno un proprio livello a seconda della tipologia di disegno: parti fisse degli sfondi, oggetti da animare e personaggi animabili. Il vantaggio di tale suddivisione consiste nel poter modificare separatamente ciascun oggetto lasciando inalterati gli elementi presenti nei layer restanti. Inoltre, soprattutto per quanto riguarda gli elementi che articolano gli sfondi, questi sono stati posizionati su layer a diverse distanze sull'asse z, in modo da simulare la profondità e in alcuni casi l'effetto di parallasse quando viene usato il movimento di camera di tipo *track*. Infatti, grazie alla funzione *Orbit* di cui dispone Moho Pro 12, è possibile visualizzare la disposizione dei vari layer nello spazio tridimensionale; in figura 3.5 è riportato un esempio di come appare la prima scena del cortometraggio in questa modalità.

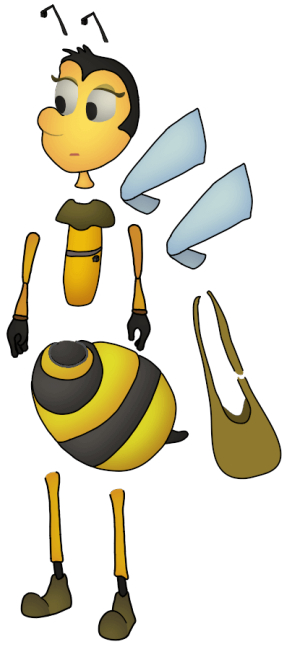


Figura 3.4: Suddivisione in layer di Pinnie

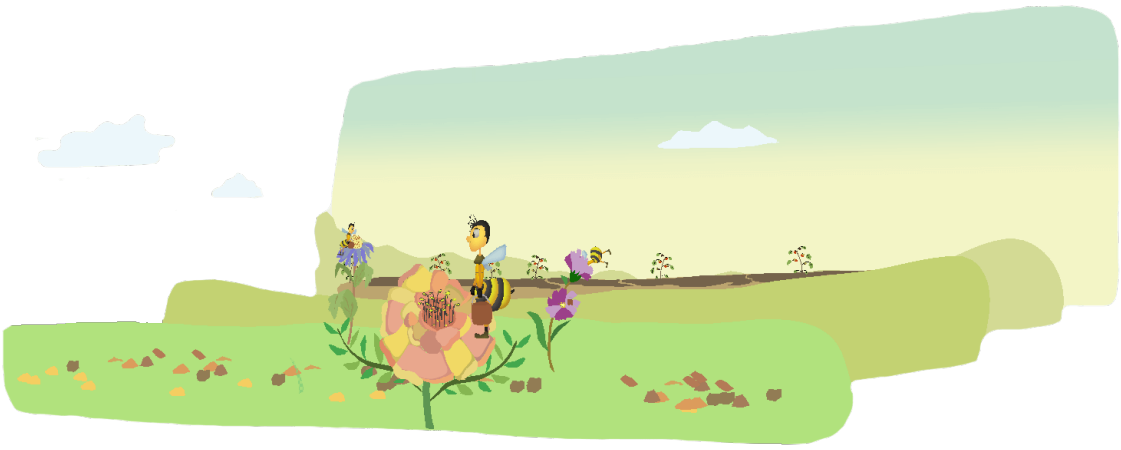


Figura 3.5: Prima scena del cortometraggio visibile in modalità Orbit.

Si può notare come gli elementi su cui si focalizza l'attenzione sono stati disposti in

primo piano e sono quelli che sono stati animati, mentre i layer contenenti gli elementi fissi che caratterizzano l'ambientazione sono stati posizionati principalmente dietro ai personaggi.

Colorazione

Nel caso di produzioni tradizionali animate in 2D, la fase di colorazione può costituire un elemento di notevole rilevanza per quanto riguarda il budget di produzione. Infatti devono essere considerati il numero di dettagli e il numero di colori utilizzati, in quanto l'utilizzo di un ampio numero di colori comporta un maggiore costo di produzione.
Occorre valutare con attenzione la scelta della palette di colori anche in base al tipo di illuminazione della scena prevista per evitare di creare sovrapposizioni di tonalità che rendono la scena complessivamente poco chiara.

Il colore è visto anche come esperienza sensoriale ed emotiva, come scrisse Vasilij Vasil'evič Kandinskij nel 1926, nel suo trattato *Lo Spirituale nell'Arte*:

> "Il colore è un mezzo che consente di esercitare sull'anima un'influenza diretta. Il colore è un tasto, L'occhio il martelletto che lo colpisce, l'anima lo strumento dalle mille corde. [...] L'arte oltrepassa i limiti nei quali il tempo vorrebbe comprimerla, e indica il contenuto del futuro."

Pertanto nella ricerca stilistica da adottare occorre tenere in considerazione il target di riferimento e quali scelte grafiche sono in grado di valorizzare la concretizzazione di un'idea. È quindi necessario porre attenzione sulla scelta dei colori che si intende utilizzare in una rappresentazione in quanto questo aspetto, oltre a definire lo stile, influisce anche sul significato narrativo percepito dallo spettatore.
Nella fase di creazione della palette di colori, occorre considerare alcuni principi che devono essere utilizzati al fine di ottenere un risultato visivo caratterizzato da equilibrio e coerenza con l'atmosfera che si vuole ricreare e proporre.
Solitamente le tonalità pastello conferiscono alla rappresentazione uno stile più morbido e coinvolgente, mentre i colori primari puri esprimono caratteristiche più particolari.
Nel libro *How to make animated films* [3], l'autore, Tony White, riporta quattro consigli su come combinare al meglio i colori per ottenere un buon risultato:

- quando si necessita di rappresentare elementi in lontananza o inquadrature prospettiche occorre delineare ciascun livello che costituisce lo sfondo variando il più possibile le tonalità dei colori. Ad esempio, i layer posti in primo piano possono essere caratterizzati da colori più scuri, quelli al centro possono invece essere definiti da colori intermedi ed infine ai livelli più in lontananza possono essere associate delle tonalità più leggere e trasparenti. In questo modo, o

utilizzando l'approccio inverso, si ottiene una sensazione di profondità ben definita.

- Siccome l'occhio umano è solitamente attirato dal punto di una rappresentazione in cui si concentra il maggior contrasto di luce, per fare in modo che gli elementi che verranno animati ottengano l'attenzione occorre che tale contrasto di colori chiari e scuri siano posizionati proprio nel punto in cui avviene l'azione principale.
- Collegato al precedente punto, per far risaltare i personaggi rispetto agli sfondi occorre che vi sia un contrasto di colori tra questi due elementi, ossia se il personaggio è definito da colori chiari di norma deve essere sovrapposto a sfondi dai colori scuri e viceversa.
- Nelle scene illuminate dal cielo, le parti verticali tenderanno ad apparire più scure, mentre le parti orizzontali risulteranno più leggere.

Pertanto, anche in questo caso sono stati realizzati due *color models*, uno che raggruppa tutti i personaggi presenti nel cortometraggio, definendo per ciascuna figura le informazioni riguardanti i colori che la caratterizzano, figura 3.7, mentre il secondo contiene le ambientazioni con gli elementi che le definiscono. Per entrambi i modelli si evidenziano inoltre il tipo di tratto utilizzato per i contorni delle forme e la qualità del tratto, ossia lo spessore.

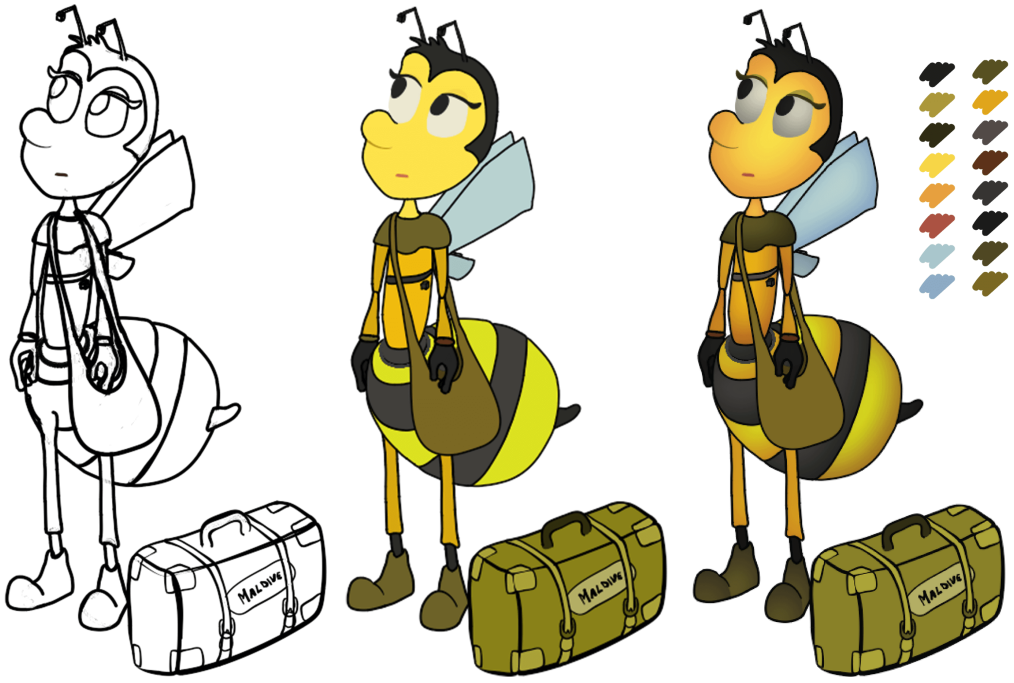


Figura 3.6: Digitalizzazione, colorazione ed effetto Shaded del disegno di Pinnie.

Per la colorazione dei personaggi, si è scelto di aggiungere particolari ombreggiature ai colori di base utilizzati in modo da creare un leggero distacco dalle ambientazioni ed evidenziare la loro presenza. Questo tipo di effetto sfumato è stato creato tramite lo strumento a disposizione *Shaded*, effettuando varie prove con diversi valori di colore, intensità e angolazione dell'ombreggiatura in base alla forma del personaggio.
In alcuni casi tale effetto è stato utilizzato anche in altri elementi presenti nelle ambientazioni, al fine di risaltare ulteriormente questi elementi.

Nella figura 3.6 vengono riassunti i tre passaggi eseguiti nella realizzazione di un personaggio: la digitalizzazione del disegno solamente con i contorni, la colorazione delle diverse shape di cui è articolato il personaggio con colori basilari ed infine l'aggiunta dell'effetto Shaded ai colori base.

Per quanto riguarda la colorazione degli elementi che costituiscono gli sfondi si è scelto di usare colori di tipo *plain* ed evidenziare alcune peculiarità di tali figure mediante l'utilizzo di shape dalle tonalità congruenti con quelle di base. Gli elementi come il cielo e l'acqua sono stati colorati usando l'effetto *Gradient* scegliendo tre diverse tonalità di colori e combinandole in proporzioni tali da ricreare l'effetto sfumatura. Nelle figure 3.8, 3.9 e 3.10 sono riportate alcune delle ambientazioni realizzate per il cortometraggio e alcune di queste immagini sono combinate con i personaggi come nelle scene effettive del corto, in modo da evidenziare il contrasto tra i colori utilizzati.

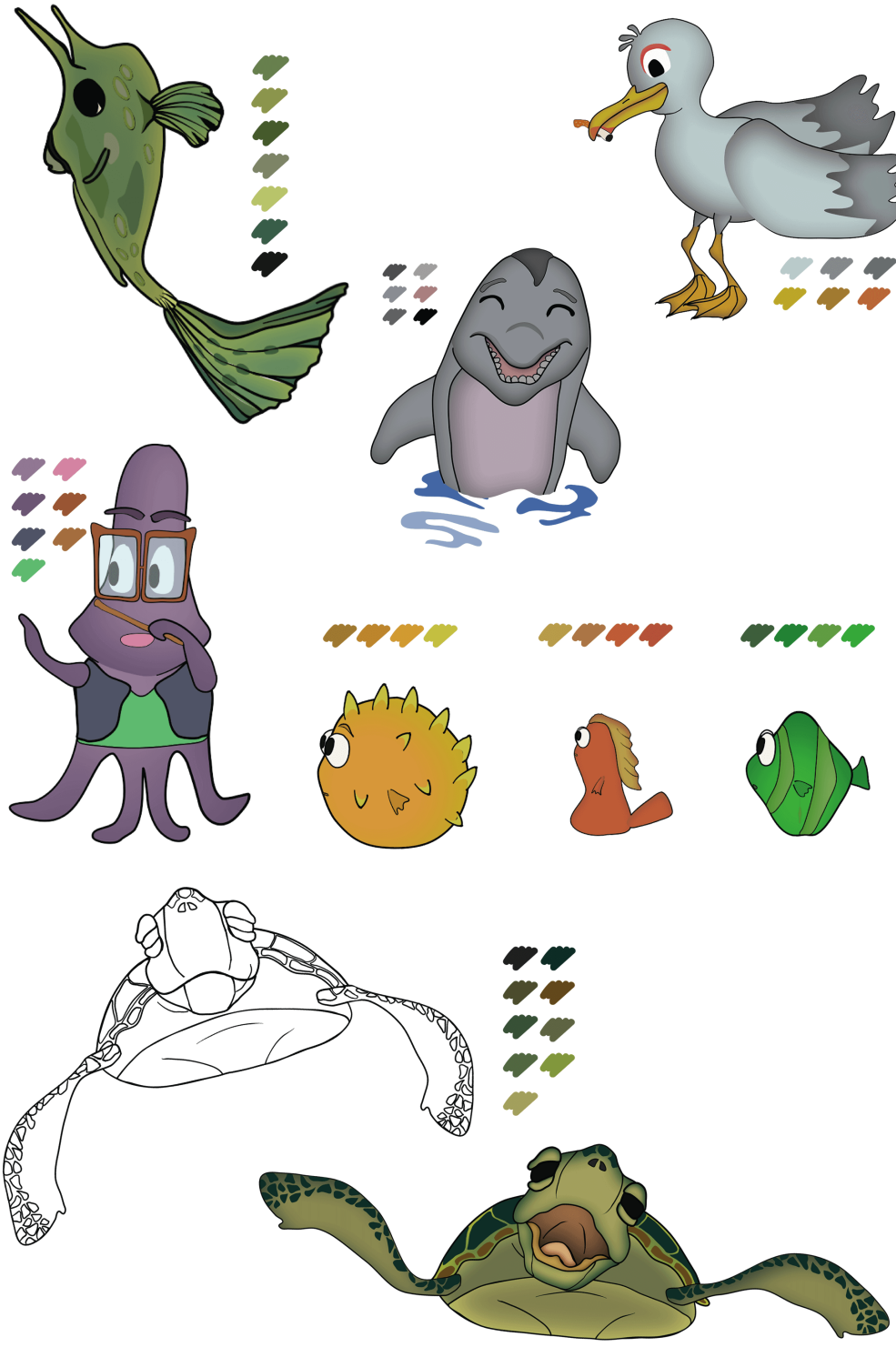
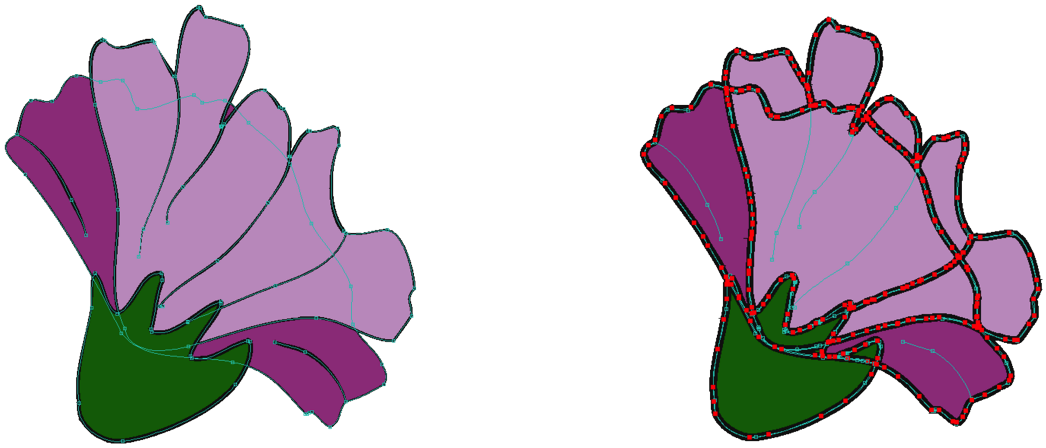


Figura 3.7: Color model di alcuni personaggi del cortometraggio

Uno degli aspetti limitanti riscontrati nell'utilizzo di Moho Pro 12, relativamente al metodo di colorazione, riguarda il tool del riempimento delle forme. Ciascuna forma chiusa che viene disegnata è composta da un'area, interna, e da un contorno che delimita e racchiude tale forma; queste due parti possono essere colorate separatamente scegliendo i due rispettivi colori prima di iniziare a tracciare la figura. Nel caso in cui si volesse successivamente cambiare colore nella parte interna della forma, utilizzando il tool *Paint bucket* questo darà origine ad una seconda forma sovrapposta a quella iniziale aggiungendo un gran numero di vettori sovrapposti, che rendono molto difficoltoso e poco piacevole l'utilizzo di tale forma quando si intende modificarla, figura 3.11. Per evitare tale problema è possibile selezionare la forma attraverso il comando *Select shape* e modificare singolarmente il colore della parte interna o del contorno attraverso la finestra *Style*.



(a) Basic shapes **(b)** Overlapping shapes

Figura 3.11: In (a) esempio di forme basiche, in (b) la stessa forma con Paint bucket.

Successivamente alla colorazione dei disegni è stata effettuata la fase di rigging dei personaggi che viene descritta nel paragrafo successivo.

Figura 3.8: Color sheet del mondo delle api.

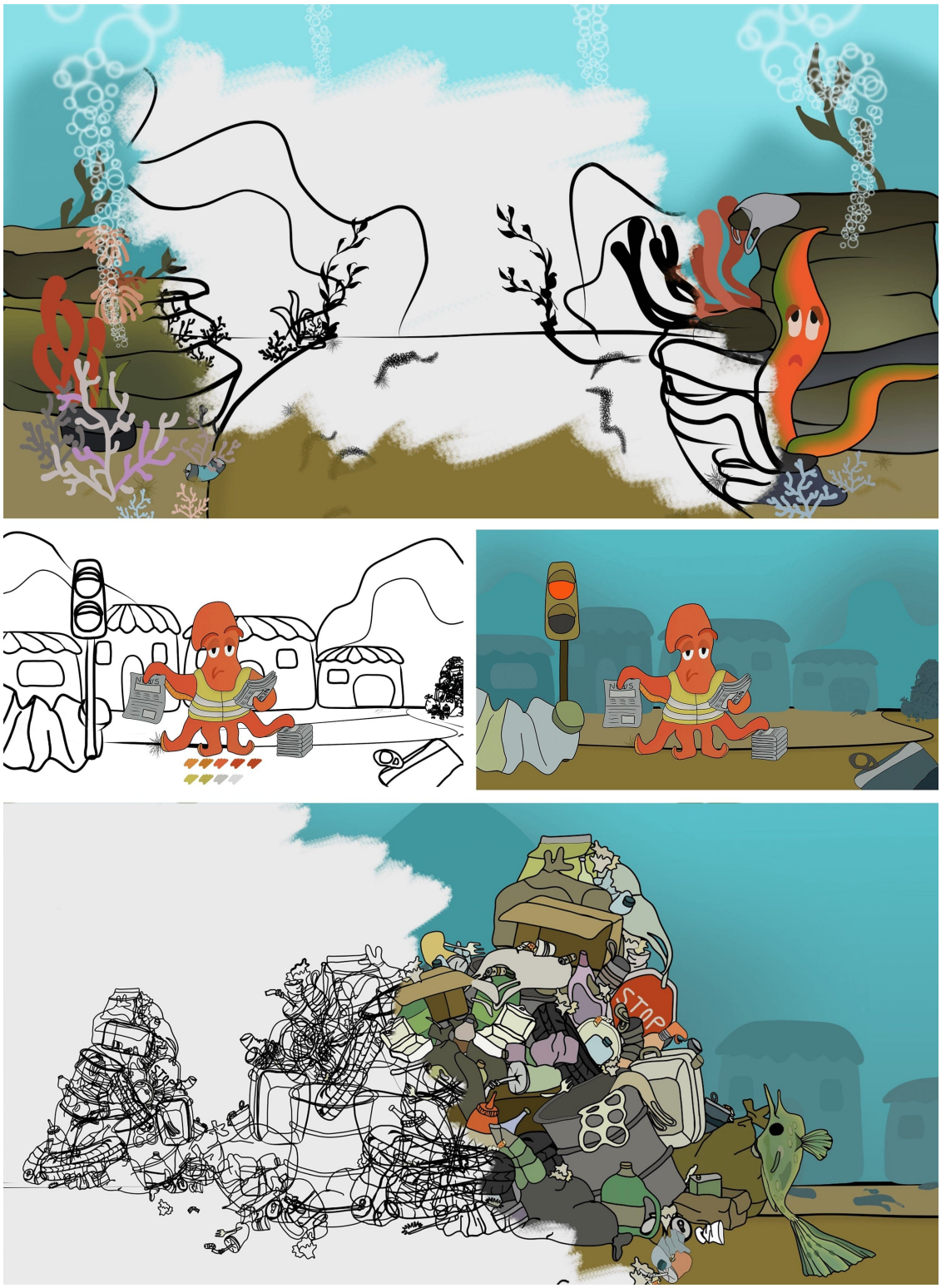


Figura 3.9: Color sheet dell'Oceano.

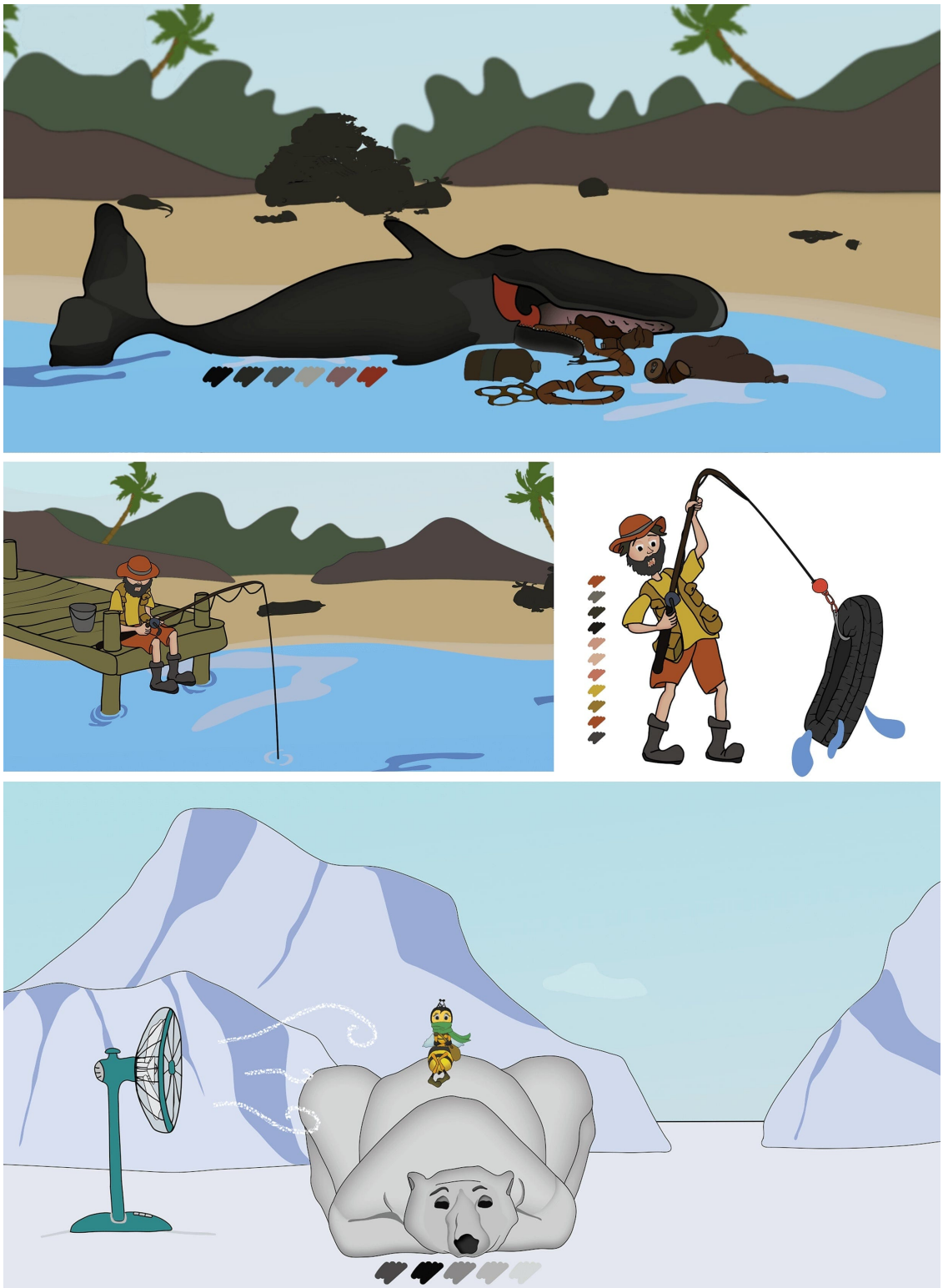
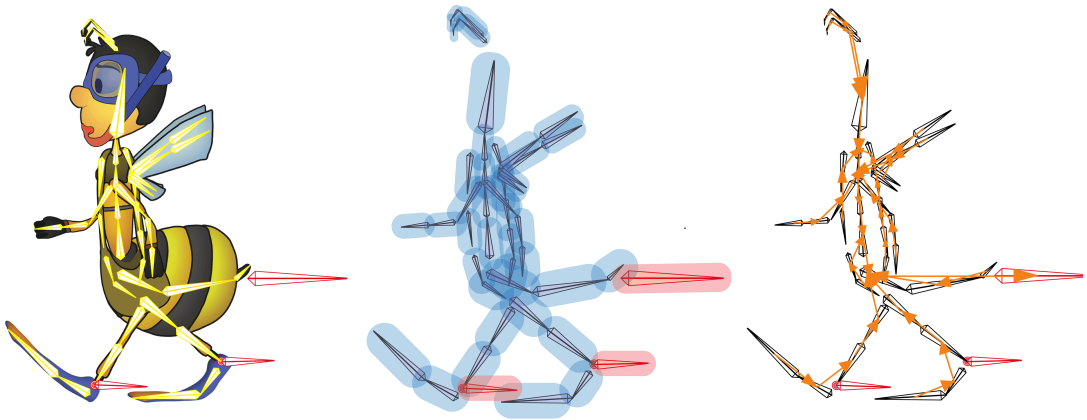


Figura 3.10: Color sheet della superficie dell'Oceano e dell'Artide.

3.3 Rigging

Il rigging è un passaggio fondamentale per eseguire la successiva fase di animazione, in quanto consiste nella preparazione strutturale del personaggio per essere animato e richiede una particolare attenzione poiché, se la struttura del personaggio non viene realizzata adeguatamente, ciò comporterà dei problemi in fase di animazione. In alcuni casi è preferibile definire il rigging dei personaggi dopo aver realizzato lo storyboard, poiché consente di avere una maggior chiarezza nei movimenti che i personaggi devono compiere, facilitando così la fase di produzione.
Nello specifico, il rigging consiste nella costruzione di una sorta di scheletro che segue la fisionomia del personaggio precedentemente disegnato.
Di seguito vengono descritti i passaggi eseguiti al fine di realizzare la struttura dei personaggi, prendendo come riferimento la figura di Pinnie.



(a) Bones e Target Bones **(b)** Bones Strength **(c)** Parent Bones

Figura 3.12

La manipolazione iniziale delle ossa in fase di creazione della struttura è possibile effettuarla solamente quando nella timeline il frame è settato a 0.
Inizialmente tutti i layer che costituiscono la fisionomia di Pinnie sono stati raggruppati in un nuovo layer, di tipo *Bone*. Quest'ultimo consente la creazione delle ossa costituenti lo scheletro del personaggio che può essere quindi manipolato come un oggetto unico. In questa fase vengono quindi impostati i controlli che verranno successivamente animati.

Selezionando il layer Bone sono state create, mediante il tool *Add bone*, le tre ossa di tipo *Target*, evidenziate di rosso in figura 3.12a. Ogni osso creato viene automaticamente nominato da Moho partendo di default dall'etichetta B1.
In seguito è stato creato un osso alla volta seguendo la sagoma del personaggio in quantità tale da ricoprire l'intera figura, e suddividendoli in base ai punti in cui era

necessario realizzare dei piegamenti, considerando inizialmente le ossa principali da cui dipendono le successive ossa secondarie come in una struttura gerarchica.
Per assegnare alle tre ossa il profilo Target, è stato prima selezionato l'osso a livello del busto e tramite la finestra *Bone Constraints*, nell'opzione della scelta Target è stato assegnato il nome dell'osso a cui far riferimento. Lo stesso procedimento è stato eseguito anche per le due ossa presenti in prossimità delle caviglie prendendo in considerazione le ossa all'altezza delle gambe.
A livello pratico, le tre ossa così definite hanno la funzione di controllare il movimento in punti specifici. In particolare, selezionando e muovendo l'osso a livello del bacino ad esempio, il personaggio si può muovere in tutte le direzioni avendo però come punti fissi le caviglie. Pertanto è utile quando si necessita di posizionare il personaggio nella posa con le gambe piegate senza muovere una gamba alla volta, creando così ad esempio una posa preparatoria per un salto; tale posa si ottiene quindi muovendo l'osso di riferimento verso il basso e automaticamente le gambe si piegheranno a livello delle ginocchia. Muovendo invece solamente le ossa Target a livello delle caviglie è possibile posizionare una gamba alla volta piegandola indipendentemente dal resto del corpo.
Restando nella finestra Bone constraints è stato attivato il parametro *Squash and stretch scaling*, una funzione fondamentale in quanto prevede proprio l'applicazione del principio Squash and Stretch, che rende 'elastica' ciascuna parte del corpo ricreando le adeguate deformazioni delle parti del personaggio durante i movimenti, mantenendo i propri volumi quando vengono scalate e ottenendo così un risultato più realistico durante l'animazione. Nello specifico, l'altezza dell'oggetto selezionato diminuirà con l'aumento della lunghezza dell'osso e viceversa, l'altezza dell'oggetto aumenterà con la riduzione della lunghezza dell'oggetto; tali deformazioni possono essere controllate impostando il fattore di scala nel riquadro disponibile accanto alla funzione.

Alle ossa di tipo Target è stato assegnato un valore nullo tramite il tool *Bone strength* in modo che l'area intorno a queste ossa non fosse influenzata dal movimento; si è quindi selezionato l'osso e tramite lo spostamento del cursore verso destra o sinistra si è ingrandita o rimpicciolita tale area. Le restanti ossa dell'armatura invece sono state selezionate e sempre mediante il tool Bone strength è stata variata l'area di copertura, evidenziata di blu in figura 3.12b, di un valore compreso tra 0,07 e 0,05; un valore superiore a tale range va ad incidere sulle altre parti del corpo non permettendo un corretto movimento di ciascun osso, d'altra parte un valore inferiore in prossimità dello 0, non consente alcun movimento delle ossa.

Durante la costruzione dell'armatura, Moho crea automaticamente l'imparentamento tra le ossa a seconda dell'ordine in cui vengono aggiunte. Nel caso in cui ci fossero degli errori che rendono aggrovigliato il movimento delle parti di cui è costituito il personaggio, questi si possono correggere tramite il tool *Reparent Bone*: occorre selezionare l'osso di cui si intende cambiare l'imparentamento, cliccando quindi sul

tool compaiono delle frecce che indicano a quale osso è imparentato, come in figura 3.12c; infine selezionando il nuovo osso a cui si vuole associare l'imparentamento la freccia cambierà direzione indicando la nuova parentela.

Al fine di ottenere la corretta associazione tra osso e parte del corpo corrispondente, nonché il movimento effettivo della figura, è stato preso in considerazione un layer alla volta contenente una parte del corpo. Quindi, sono state successivamente selezionate le ossa che rientravano nella forma presente sul layer e tramite la finestra *Bone* si è attivata l'opzione *Use Selected Bones For Flexi-Binding*, selezionando successivamente la funzione *Bind Layer* dalla finestra dei tool disponibili. In questo modo, andando sul layer principale di tipo Bone si è provato a muovere l'osso appena associato, tramite il tool *Manipulate Bones* che in questa fase, al frame 0, assume la funzione di test del movimento temporaneo del personaggio, ossia consente di verificare se i passaggi eseguiti nell'associazione ossa-layer permettono di eseguire il giusto movimento del personaggio. Infatti appena l'osso verrà deselezionato, la posa del personaggio ritornerà nella posizione di default creata senza memorizzare alcun movimento nella timeline.
Oltre allo strumento Bind Layer utilizzato, Moho dispone anche di un secondo tool detto *Bind Point* che prevede una modalità diversa di associare le ossa alle shape disegnate. Nello specifico, il tool funziona con lo stesso principio di Bind Layer ma invece di agire sui layer, in questo caso occorre selezionare un gruppo di punti che costituiscono le varie shape del personaggio per collegarle ad un determinato osso. Pertanto cliccando sul tool Bind Point, automaticamente verrà creato il collegamento.

Dopo aver impostato la struttura ossea del personaggio si è effettuata la fase di animazione effettiva che prevede la manipolazione del personaggio tramite il tool Manipulate Bone, precedentemente citato, il quale ora assume la funzione di manipolatore permanente, ossia consente di muovere le ossa e di conseguenza di creare una posa che verrà memorizzata nella timeline tramite dei keyframe; questo solamente quando sulla timeline si è posizionati su un frame successivo allo 0. Inoltre si ha la possibilità di modificare le ossa durante l'animazione tramite il tool *Transform Bone*, uno strumento che consente di traslare, ruotare o scalare un osso selezionato. Quando viene selezionato questo tool, compaiono due punti alle estremità dell'osso che indicano il tipo di trasformazione che si può applicare in base a dove viene posizionato il cursore.

La fase di animazione verrà trattata in dettaglio nel paragrafo successivo riportando elementi teorici e alcuni esempi pratici che sono stati realizzati.

3.4 Animazione

Per comprendere meglio il concetto di animazione ed i principi che ne derivano, occorre specificare il significato di quelle che Chris Webster indica come "le 4 A dell'animazione" [4], le quali, rappresentate sotto forma gerarchica, indicano i vari livelli di animazione che si possono ottenere. Di seguito vengono spiegati a partire dal livello più basso della gerarchia:

- Activity: si trova al livello più basso della gerarchia e rappresenta la categoria che descrive la forma più semplice di movimento esistente. Tali movimenti non possono essere associati a qualsiasi cosa che si trova in natura e pertanto sono astratti. L'azione può essere rappresentata ad esempio da un'immagine che inizialmente si trova in un particolare punto su uno schermo e successivamente tale immagine la si ritrova in un altro punto in un secondo momento. Un ulteriore esempio è dato dallo scorrimento del testo in una sequenza di titoli. Anche se alcuni oggetti vengono visualizzati come elementi che presentano diverse dinamiche di movimento in vari punti, rimangono comunque astratti e non possono essere associati a particolari oggetti che possiamo riconoscere come capaci di muoversi in modo indipendente.

- Action: descrive il movimento che può essere attribuito ad oggetti specifici, ad esempio il rimbalzo di una palla o l'azione di un aeroplanino di carta. La differenza tra attività e azione è data dal fatto che nell'azione di un oggetto noto riconosciamo un determinato movimento che può essere associato ad un oggetto comune, il quale si muove in condizioni a noi note ed è soggetto alle conosciute leggi della natura. Ad esempio riconosciamo il movimento del vento che soffia tra le foglie di un albero o il movimento delle onde del mare, ossia movimenti che esistono in natura e per via delle loro proprietà particolari.

- Animation: rappresenta il tipo di dinamica che ha origine all'interno della materia. Un esempio può essere il movimento di un salmone che salta fuori dall'acqua o un colibrì che raccoglie il nettare da un fiore. L'animazione è quindi caratterizzata da questi movimenti che derivano proprio dalle azioni compiute dal soggetto stesso; il modo in cui avvengono tali movimenti sono dati dalla fisionomia e da particolari motivazioni che suscitano determinate azioni.

- Acting: rappresenta il più alto livello del movimento animato. I movimenti non sono quindi soggetti alle sole leggi della natura e ad azioni che provengono dalle motivazioni interiori del soggetto, ma esistono determinate ragioni psicologiche che determinano questi movimenti, ossia le performance. In questa tipologia di movimento vengono analizzati i sentimenti interiori del soggetto dell'animazione; pertanto è possibile non solo vedere l'azione che compie il

soggetto ma captare anche le emozioni che prova in quel determinato istante. Tale categoria della gerarchia è considerata come quella più complessa da affrontare, ma essenziale per creare narrazioni coinvolgenti.

In tale contesto trovano quindi spazio i cosiddetti principi dell'animazione, trattati in dettaglio nel sottoparagrafo successivo.

3.4.1 I principi dell'animazione

Nei primi anni dalla nascita degli studi Disney nel 1923, gli animatori iniziarono ad utilizzare termini come *aiming*, *overlapping* e *pose to pose* che suggerivano la necessità di associare un determinato nome ad una particolare procedura di animazione.

> "I verbi si sono trasformati in sostantivi durante la notte, come, ad esempio, quando il suggerimento 'Why don't you stretch him out more?' è diventato 'Get more stretch on him'. 'Wow! Look at the squash on that drawing!' non significava che una verdura si fosse spiaccicata su un'opera d'arte; questo indicava che qualche animatore aveva mostrato con successo un personaggio in una postura appiattita. Alcune di queste terminologie assegnavano solo nuovi significati a parole familiari e convenienti. Fare una scena potrebbe significare rappresentare i movimenti voluti, o addirittura animarli; è una volta che è stata fatta, la scena è passata al reparto successivo."[5]

Gli animatori avevano bisogno di ricercare metodi di disegno e animazione da poter adattare a qualsiasi personaggio od oggetto e rendere il movimento più realistico basandosi sulle leggi della fisica, nonché creare un linguaggio comune tra gli animatori per avere una maggior sicurezza e una più attendibile previsione sul risultato finale. Da qui ebbero origine i 12 fondamentali principi dell'animazione che vengono descritti di seguito.

Squash and Stretch

Squash and Stretch è considerato il più importante fra i principi dell'animazione. Mediante tale principio, sia i personaggi che gli oggetti rigidi possono essere deformati cambiando le proprie masse e volumi. La deformazione ha un ruolo primario soprattutto nell'animazione facciale in quanto permette di evidenziare particolari caratteristiche del viso contribuendo ad una mimica facciale efficace.
Durante l'applicazione di questo principio, bisogna tenere in considerazione alcuni vincoli in modo da ottenere un buon risultato. Quando un oggetto, ad esempio

una palla, viene schiacciata in una certa direzione, occorre che venga contemporaneamente allungata nella direzione opposta e viceversa, questo per poter mantenere costante il volume, altrimenti si otterrà un risultato in cui l'oggetto apparirà solamente ingrandito o rimpicciolito senza alcun effetto.

In figura 3.13 vi è un esempio estratto dal cortometraggio di animazione rappresentante alcune pose del pesce palla a cui è stato applicato il principio di Squash and Stretch:

a. Il pesce palla è nella posizione iniziale senza alcun effetto.

b. Il pesce palla è stato prima 'stirato' in verticale e poi 'schiacciato' in orizzontale prima di toccare nuovamente il sedile.

c. Il pesce palla è stato prima 'schiacciato' in verticale e poi 'stirato' in orizzontale, toccando il sedile.

d. Il pesce palla torna nella posa iniziale senza alcuna deformazione.

e. Il pesce palla è stato ingrandito uniformemente nello spazio bidimensionale per simulare il movimento tipico di questa tipologia di pesce.

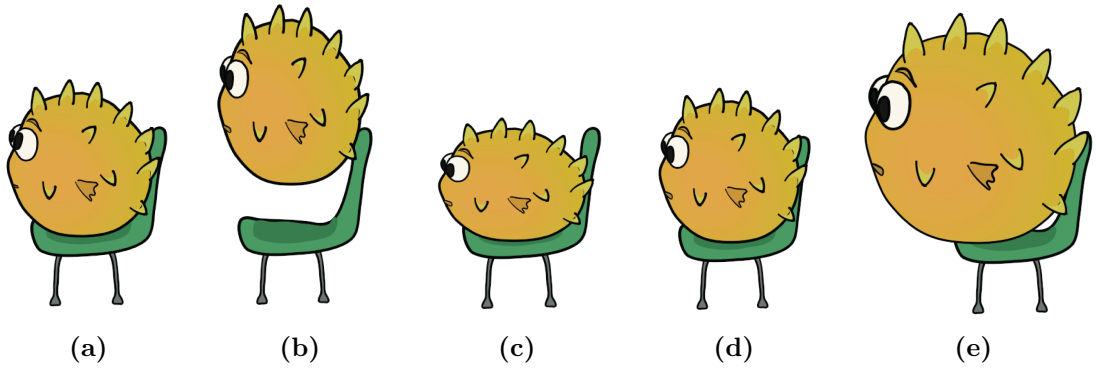


Figura 3.13: Esempio del principio Squash and Stretch tratto dal cortometraggio.

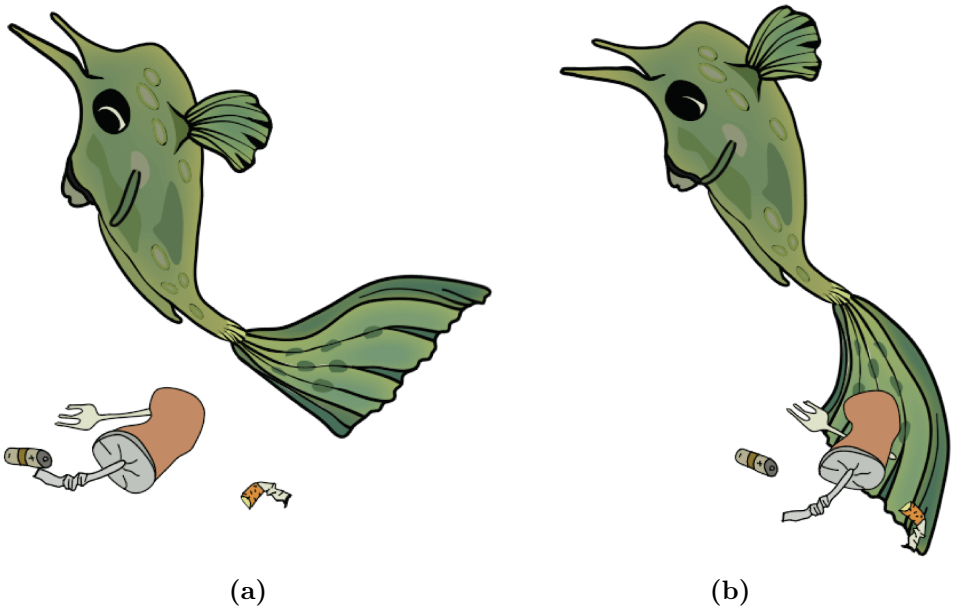
Anticipation

Il principio dell'anticipazione indica la preparazione di un'azione. Nello specifico si tratta di un movimento che in genere avviene rapidamente per suggerire allo spettatore l'azione vera e propria che avverrà successivamente. Tale movimento avviene nella direzione opposta all'azione effettiva e se svolta lentamente consente di evidenziare maggiormente l'azione principale. Ad esempio nel caso di un calcio al pallone, il personaggio inizialmente porta lentamente la gamba indietro e poi calcia rapidamente in avanti il pallone.
L'anticipazione può anche essere utilizzata a livello narrativo ad esempio quando si

intende spostare l'attenzione su un punto diverso, il personaggio che inizialmente guarda in avanti, poi gira lo sguardo verso sinistra per annunciare che una nuova azione arriverà da quella direzione.
Un esempio di anticipazione è rappresentato in figura 3.14 che mostra due frame in cui il personaggio Fish è intento a raccogliere i rifiuti per spostarli in avanti:

a. Inizialmente la coda del pesce è ruotata nella direzione opposta ai rifiuti ed il busto e la testa sono incurvati in senso orario.

b. La coda è ruotata nella direzione dei rifiuti per raccoglierli mentre il busto e la testa sono ruotati in senso antiorario, anch'essi nella direzione dei rifiuti.



(a) **(b)**

Figura 3.14: Esempio del principio Anticipation tratto dal cortometraggio.

Staging

Lo *Staging* consiste nella messa in scena, ossia nel posizionamento del soggetto che compie l'azione principale in un determinato punto dell'inquadratura in modo che lo spettatore possa soffermarsi su tale soggetto catturando tutti i suoi movimenti.
Il personaggio viene quindi posizionato a seconda di dove si intende guidare la vista dello spettatore ed occorre concentrarsi su un dettaglio alla volta. Ad esempio è anche possibile che all'interno di una scena in cui ogni elemento si muove tranne uno, l'attenzione ricada proprio su quest'ultimo; in questo caso si evidenzia un'idea per contrasto.
Il modo in cui posizionare un determinato elemento deve in ogni caso seguire la

regola dei terzi, ossia immaginare di suddividere l'inquadratura in nove riquadri rettangolari eguali e concentrare il fuoco dell'azione in un punto dato dalla cornice; può essere ad esempio al centro o in uno dei quattro punti interni determinati dalla suddivisione dell'inquadratura. Come si nota nella figura 3.15, in questa scena si è deciso di concentrare l'azione principale di Pinnie intenta a raccogliere il polline nel rettangolo centrale ed in quello sottostante, nello specifico sulla destra dei punti di delimitazione del riquadro centrale; invece le azioni secondarie delle due api si svolgono nei rettangoli periferici.

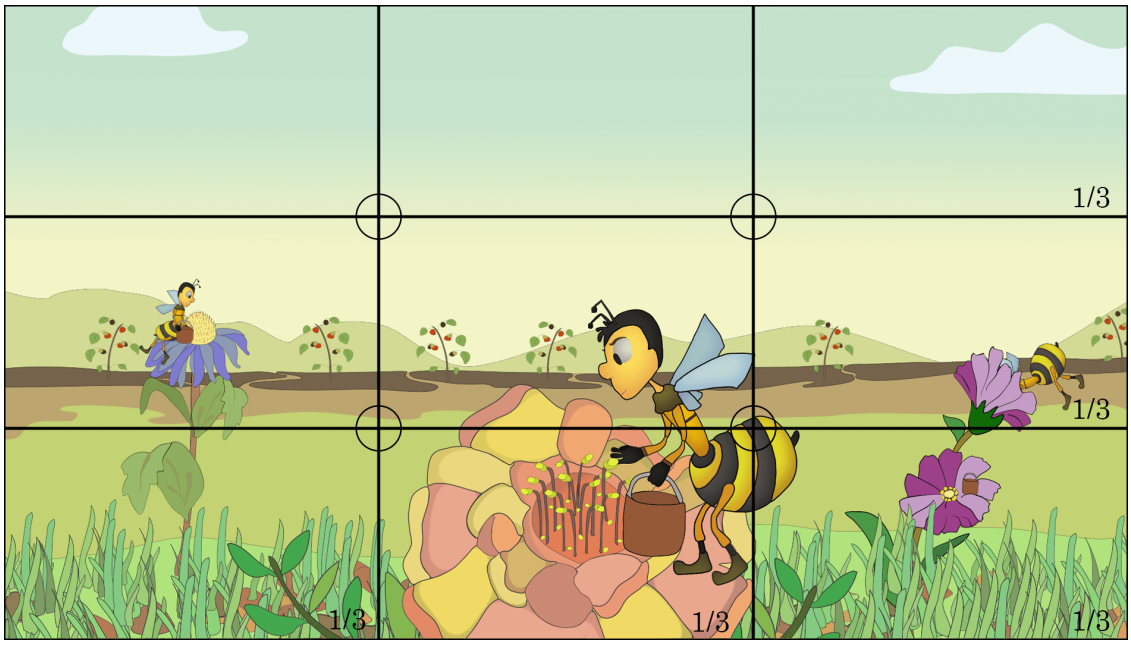


Figura 3.15: Esempio del principio Staging tratto dal cortometraggio.

Esistono anche ulteriori modalità di Staging che però sono meno diffuse in quanto vengono considerate degli stereotipi, tra cui il posizionamento del soggetto principale proprio nel riquadro centrale. Oltre al posizionamento degli elementi nello spazio, per focalizzare l'attenzione è possibile utilizzare colori e luci combinati in modo da evidenziare determinati punti rispetto ad altri.

Straight Ahead Action and Pose to Pose

Il principio *Straight Ahead action and Pose to Pose* si riferisce a due diversi metodi di creare un'animazione.
Straight Ahead indica la creazione del movimento in maniera diretta, ossia viene animato in sequenza fotogramma dopo fotogramma fino alla fine dell'azione. Tale metodo permette di ottenere un movimento finale fluido ma risulta più complesso

poiché non è prevedibile, infatti può creare problemi soprattutto nell'animazione 2D, in quanto è possibile che involontariamente si modifichino le proporzioni e le dimensioni del personaggio da animare.

Pose to pose è invece il metodo che prevede la realizzazione del movimento indicando inizialmente tutte le pose chiave, *keyframe*, ossia la posizione iniziale, il movimento centrale, *braking pose*, e la posa finale, *extreme pose*; successivamente vengono definite ulteriori pose chiave dette *passing key* prima di inserire i rimanenti *in between*, ossia le pose intermedie. Tale procedura consente un maggior controllo dell'animazione ma risulta poco fluida.
Pertanto è consigliabile utilizzare una combinazione di entrambi i metodi descritti per ottenere un risultato più efficace.

Follow Through and Overlapping Action

Questo principio indica l'azione che viene rappresentata nell'istante successivo alla conclusione dell'azione principale. Nello specifico, il *Follow Through* definisce la parte finale dell'azione; ad esempio nell'animazione del lancio di una palla, la mano del soggetto che la lancia continua a muoversi anche dopo aver lasciato la palla. Oppure nel caso di una camminata, tale approccio tiene in considerazione il movimento delle singole parti del corpo del personaggio in questione. Si può ad esempio prendere in considerazione il bacino del personaggio come guida rispetto alle restanti parti del corpo; quando il bacino si muove, la gamba segue tale movimento, mentre quando si ferma, gli altri elementi del corpo continuano a muoversi con leggero ritardo.
L'*Overlapping Action* agisce proprio sulla differenza delle masse degli elementi che costituiscono un personaggio, ossia il movimento di tali parti avviene in tempi diversi a seconda della massa; parti più pesanti ritardano il movimento e si arrestano più lentamente. Infatti solitamente le parti più esterne dal tronco di un personaggio tendono ad essere più leggere limitandosi a seguire i movimenti prodotti dalla parte centrale del corpo.
Un esempio tratto dal cortometraggio si può notare nel movimento delle antenne di Pinnie, le quali vengono considerate come elementi 'follower' che seguono l'azione principale della testa e nel movimento delle ali che si muovono in ritardo rispetto al busto. In figura 3.16 sono visibili tre pose in cui si evidenzia tale principio:

a. le spalle e la testa di Pinnie sono alzate, contemporaneamente anche le ali e le antenne sono state disegnate in direzione verticale per enfatizzare il momento di stupore previsto in questa scena.

b. Le spalle e la testa di Pinnie sono state abbassate lasciando inalterate le posizioni delle ali e delle antenne.

c. Nei frame successivi sono state abbassate anche le ali e le antenne evidenziando il movimento che avviene in leggero ritardo rispetto al movimento principale delle spalle e della testa.

(a) **(b)** **(c)**

Figura 3.16: Esempio del principio Follow Through and Overlapping Action tratto dal cortometraggio

Slow In and Slow Out

Il principio *Slow In and Slow Out* indica la graduale variazione di velocità fra le pose di un'azione svolta da un personaggio, la quale ha lo scopo di rendere tale azione più espressiva e realistica.
Un'azione *Slow In*, detta anche *Ease In*, descrive un movimento che decelera quando arriva in prossimità di un keyframe. Movimenti di questo tipo sono usati ad esempio quando un oggetto rallenta prima di arrestarsi. Un oggetto che ha poca massa e poco slancio perde la sua energia cinetica più velocemente di un oggetto più pesante e rallenta più rapidamente. Generalmente oggetti più leggeri rallentano più lentamente di quelli più pesanti.

Figura 3.17: Esempio di Slow In e Slow Out

Slow Out, o *Ease Out*, rappresenta invece un'azione che accelera progressivamente dopo un keyframe. Tale azione si presenta all'inizio di un'animazione di un oggetto. Maggiore è l'inerzia di un oggetto, maggiore sarà il tempo necessario a definire lo slancio che può portare ad un'azione di slow out più pronunciata.

In figura 3.17 è illustrato un esempio di tale principio utilizzato per realizzare il volo delle api, dalla fase di preparazione dello slancio fino al volo effettivo. In particolare la prima ape di destra rappresenta la posa iniziale di base, la seconda ape con la valigia blu si prepara allo slancio e tale anticipazione rappresenta la fase di slow in che prevede un maggior numero di frame ravvicinati, per rendere il movimento lento, infine l'ultima ape che porta la valigia verde raffigura la fase di slow out, in cui l'ape acquista la massima velocità. La rapidità di quest'ultimo movimento è accentuata dall'aggiunta dell'effetto di motion blur.

Arcs

Il principio *Arcs* consiste nel rappresentare il movimento di un'azione seguendo una traiettoria guida che simula proprio delle forme curvilinee come degli archi; ciò permette di ottenere movimenti percepiti dallo spettatore come più naturali. Nei casi in cui tale principio non viene applicato e quindi i movimenti vengono rappresentati lungo linee rette, si ottengono delle animazioni rigide, come nel caso di elementi meccanici.
In figura 3.18 è riportato un esempio di movimento ad arco utilizzato per l'animazione del movimento della tartaruga, nello specifico il movimento delle zampe all'indietro per simulare il personaggio che nuota nell'oceano.

Figura 3.18: Esempio di Arcs

Secondary Action

Per rendere l'animazione più realistica ed espressiva, occorre tenere conto del principio che prevede l'aggiunta di azioni secondarie rispetto all'elemento principale nella scena; queste azioni però non devono sovrastare l'azione principale altrimenti l'attenzione dello spettatore non sarà più incentrata sull'idea principale che si intende trasmettere. Un esempio è il caso di un personaggio che cammina mostrando una certa personalità attraverso determinati movimenti; l'aggiunta del movimento facciale che simula il fischiettare di tale personaggio può essere considerata come un'azione secondaria.

Timing

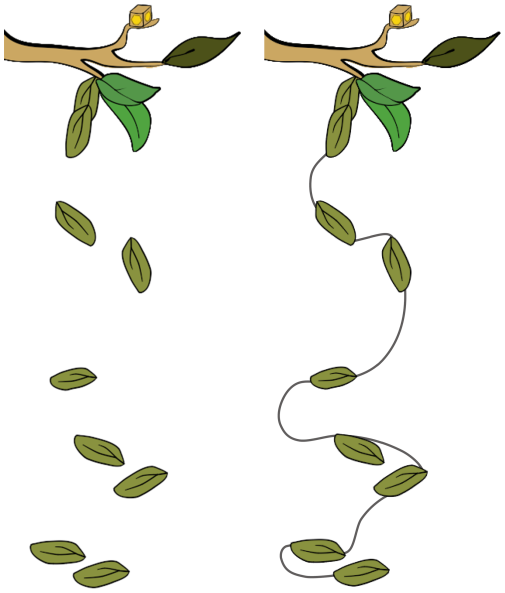
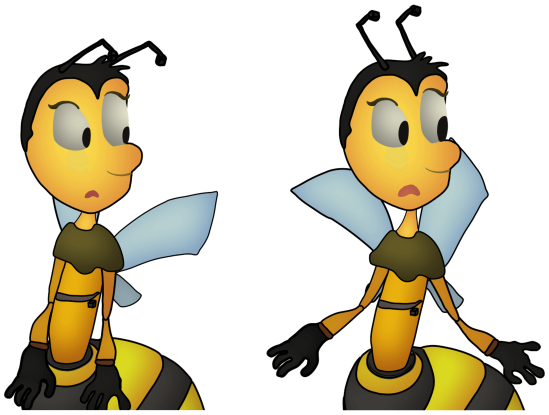


Figura 3.19: Esempio di Timing

Il principio del *Timing* indica il ritmo di un'azione, nonché la quantità di tempo necessaria da attribuire all'anticipazione di un'azione, all'atto stesso e agli effetti che ne derivano. Come accennato nel paragrafo dell'animatic, se un'azione viene compiuta in un lungo tempo ciò distoglierà l'attenzione dello spettatore, in quanto risulterà confusa, d'altra parte, una tempistica eccessivamente rapida potrebbe non essere notata e/o non compresa.
Il timing è un aspetto importante anche per evidenziare la personalità, il peso e le dimensioni di un personaggio o di un oggetto. Ad esempio un oggetto che ha un peso significativo tenderà a muoversi più lentamente, oppure a seconda della velocità con cui un personaggio muove la testa verso sinistra o verso destra si può comprendere se l'intenzione è quella di guardare a sinistra e a destra, oppure se vuole esprimere un senso di negazione, nel caso in cui il movimento sia più veloce. Per quanto riguarda le azioni secondarie, queste avvengono in tempi successivi all'azione principale. Dal punto di vista tecnico, il timing di un'animazione è determinato dalla sequenza di frame e dalla loro posizione all'interno di tale sequenza: maggiore è il numero di frame ravvicinati in una sequenza stabilita, minore sarà la velocità dell'azione; viceversa, maggiore è la distanza tra i frame nella sequenza maggiore sarà la velocità dell'azione.
Nell'azione della foglia che si stacca dal ramo, raffigurata in 3.19, ogni posa della foglia rappresenta un frame chiave della timeline. La parte iniziale dell'azione prevede quattro frame, foglie, disposte abbastanza lontane le une dalle altre per simulare la caduta rapida, mentre le restanti quattro foglie che costituiscono l'azione finale sono state posizionate più vicine le une alle altre in modo da rendere l'azione più lenta ed evidenziarla maggiormente. Inoltre la foglia è stata disposta seguendo un percorso curvilineo, disegnato appositamente come guida, per simulare l'andamento fluttuante, nonché la leggerezza della foglia.

Exaggeration

Il principio dell'esagerazione consiste nell'accentuare un concetto attraverso particolari forme o azioni, ad esempio evidenziando determinate caratteristiche di un personaggio anche a livello espressivo. Ciò conferisce all'animazione un maggior coinvolgimento da parte dello spettatore.
In figura 3.20 vi è un esempio della stessa posa di Pinnie che esprime spavento in due versioni diverse: nella prima figura l'espressione del viso e gli elementi come le antenne, le ali e le braccia sono meno marcate rispetto alla seconda posa in cui soprattutto il volto, con la bocca e le pupille degli occhi, è molto più accentuato e la testa è sollevata fin quasi al limite del collo, ottenendo un maggior effetto di caricatura.



(a) No Exaggeration **(b)** Exaggeration

Figura 3.20: Esempio di Exaggeration

Solid Drawing

Il principio del *Solid Drawing* prevede che vengano raffigurati personaggi bidimensionali in pose tali da simulare la tridimensionalità. Nella creazione dei personaggi e degli elementi di una scena occorre pertanto considerare lo spazio 3D attribuendo a tali elementi un peso ed un volume. Nel caso di un personaggio, questo deve essere ritratto da diverse angolazioni a seconda delle azioni che egli deve svolgere, rendendo così l'animazione più realistica.

Appeal

Il principio dell'*Appeal* indica la necessità di realizzare una rappresentazione piacevole alla vista per poter catturare l'attenzione dello spettatore. Occorre quindi porre attenzione alla scelta stilistica dei personaggi e delle ambientazioni, alle loro disposizioni nella composizione della scena, evitando le simmetrie, e alle azioni da rappresentare.

L'uso dei principi appena trattati sono fondamentali per ottenere un'adeguata fluidità nei movimenti ed ottenere un maggior realismo sul risultato finale.

Nei sottoparagrafi successivi verranno descritti due particolari procedimenti di animazione realizzati per il cortometraggio che prevedono, oltre all'applicazione

dei principi sopra descritti, il concetto di ripetizione di alcune pose chiave che permettono di creare un'animazione continua.

3.4.2 Walk cycle

L'animazione ciclica è un metodo efficace da utilizzare in casi in cui si necessita di realizzare azioni ripetitive come ad esempio la camminata, la corsa, il movimento di una bandiera ecc. Dal nome *cycle* si deduce quindi che si tratta di animazioni che avvengono in continuità partendo da una posa iniziale fino ad arrivare alla posa finale e successivamente si ritornerà alla posa iniziale. È preferibile utilizzare tale procedura in azioni di breve durata per evitare che sia visibile la ripetitività della posa iniziale del ciclo e quindi che svanisca l'effetto di continuità. Tale metodo è applicabile solamente nell'animazione classica 2D e nella computer animation grazie a software che permettono la ripetizione di una sequenza di frame nella linea temporale.

Nella produzione del cortometraggio sono stati realizzati due tipi di animazioni cicliche: il walk cycle di Pinnie, descritto di seguito, e il volo del gabbiano Gull trattato nel paragrafo successivo. Per realizzare il ciclo di camminata è stato preso come esempio il materiale riportato in [6].

Il walk cycle è uno degli esempi base dell'animazione che suscita l'illusione del movimento e rientra quindi nella categoria della terza A, Animation, presente nella gerarchia descritta precedentemente; pertanto l'animazione del soggetto in questione deve essere eseguita tenendo in considerazione la motivazione per cui tale azione viene eseguita dal soggetto stesso.
Per approcciarsi alla realizzazione di un ciclo di camminata che sia credibile occorre considerare diversi fattori relativi alla fisionomia del soggetto e all'aspetto psicologico che caratterizzano tale figura.
Esistono inoltre diverse reference che spiegano i passaggi standard da seguire per poter ricreare un ciclo di camminata basilare, tuttavia occorre porre particolare attenzione a tali passaggi in quanto raramente garantiscono un risultato fluido. Pertanto è necessario modificare ed adattare tali punti chiave a seconda dell'esigenza e delle caratteristiche del personaggio che si vogliono trasmettere.

La camminata è un processo ottenuto dallo spostamento delle gambe in sequenza. Più è lento il movimento delle gambe maggiore è il controllo che si ha sul movimento. In genere la camminata avviene sollevando leggermente un piede alla volta da terra.
Durante la camminata si può notare che per dare spinta ed equilibrio al movimento, ogni braccio è coordinato alla gamba opposta. Quando si scende si rilascia il massimo dell'energia e le braccia si trovano nel punto più ampio, mentre il tallone è la prima parte del piede che tocca terra. Quando invece si risale il movimento

è più lento in quanto si ha un accumulo di energia potenziale. Inoltre i polpacci esercitano una forza ogni volta che si spingono in avanti le gambe.
Tutte le camminate sono diverse le une dalle altre e dipendono da numerosi fattori sia a livello della fisionomia del soggetto in questione sia a livello della personalità e allo stato d'animo che egli trasmette.

Il ciclo di camminata di Pinnie, rappresentato in figura 3.21, è stato creato in modo da simulare la personalità decisa della protagonista intenta ad iniziare il suo viaggio e combattere per la sua causa.

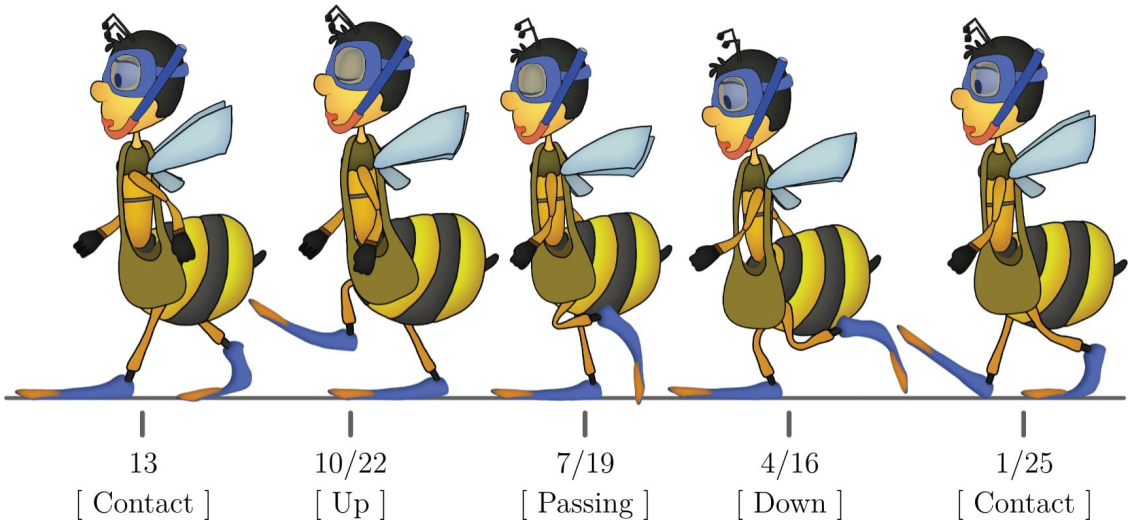


Figura 3.21: Frame corrispondenti alle pose chiave che costituiscono il walk cycle di Pinnie.

Ciò che simula il peso del soggetto è l'alternanza della posa del corpo verso il basso con la posa del corpo rialzata. In particolare, la sensazione di pesantezza del corpo è percepita quando la posa rappresenta il soggetto con le gambe piegate e di conseguenza il corpo abbassato.

Per costruire il ciclo di camminata si è iniziato dal raffigurare le due pose di contatto. Le braccia sono nella direzione opposta alle gambe per dare equilibrio e spinta.
Successivamente si è creata la posizione centrale o posizione di passaggio, *passing*, che rispetto alle altre pose è posta leggermente più in alto. In questa posizione la gamba destra è dritta per anticipare il sollevamento del bacino, del corpo e della testa.
È stata poi inserita la posizione di *down* in cui Pinnie è abbassata portando il peso del corpo sulla gamba destra, mentre il braccio destro è disteso al massimo verso il basso e in direzione opposta alla gamba destra.
Infine segue la posa di *up*, in cui ora è la gamba sinistra piegata verso l'alto mentre la gamba destra è distesa all'indietro con il piede completamente appoggiato a terra in modo da sollevare il corpo.

Quindi, il peso si trova in basso subito dopo la posa del contatto, mentre invece si trova in alto dopo la posizione di passaggio.

Il ritmo in cui avviene il ciclo di camminata è stato impostato su 24 frame.
Dopo aver definito le pose descritte nella timeline, sono stati inseriti dei frame intermedi ed aggiustati i movimenti fino ad ottenere un movimento finale il più fluido possibile. In figura 3.22 è riportata la timeline di Moho contenente tutti i keyframe creati per ottenere il ciclo di camminata di Pinnie.

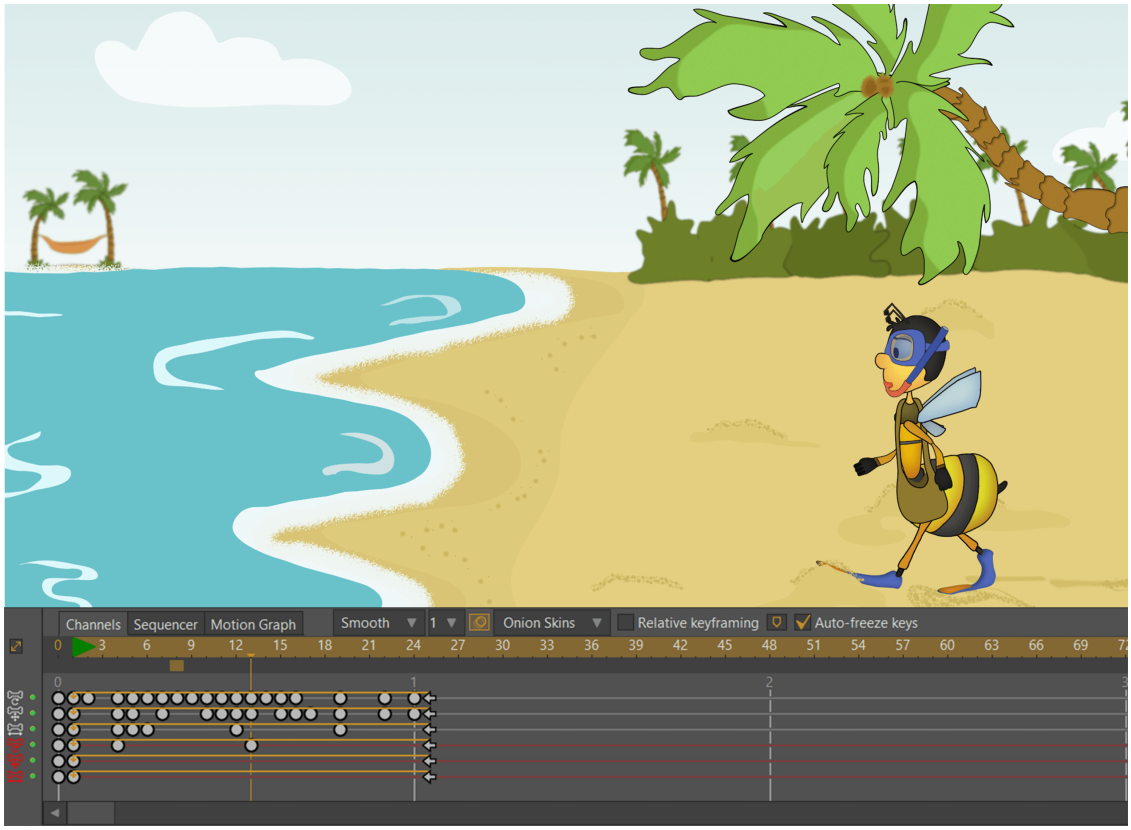


Figura 3.22: Timeline contenente i keyframe complessivi utilizzati per il walk cycle. Nello specifico la figura mostra la posa di Pinnie corrispondente al frame 13, ossia una posizione di contatto.

3.4.3 Il volo del gabbiano

Replicare il movimento degli animali affinché siano il più possibile vicini alla realtà richiede una particolare attenzione alle azioni che essi compiono, nonché un'analisi dettagliata dei loro movimenti per poter comprenderli al meglio.
L'uso di reference come fotografie e video sono uno strumento fondamentale da utilizzare come base dell'animazione.

Per realizzare l'animazione del gabbiano Gull, personaggio del cortometraggio, è stato usato come riferimento il materiale descritto nel libro *Animation. The mechanics of motion* [4].

Diverse specie di volatili presentano modi diversi di muoversi e volare e tali azioni sono determinate da caratteristiche come la dimensione, la fisionomia e l'ambiente in cui vivono: ad esempio le aquile e i falchi hanno un andamento molto rapido mentre le oche sbattono le ali lentamente.
Più il volatile è caratterizzato da una fisionomia imponente, più lentamente muoverà le proprie ali, ma sufficientemente veloce da rimanere in volo. Pertanto, il tempo di battito delle ali determina la dinamica del movimento del volatile.

Le principali fasi che costituiscono il volo di un volatile sono:

1. Take off: la fase di decollo.
2. Powered flight: la forza che permette lo slancio del corpo.
3. Acceleration: la fase in cui si accelera il movimento del corpo.
4. Cruising: ossia la fase in cui le ali vengono agitate ripetutamente le une con le altre.
5. Gliding: la fase in cui il volo avviene senza agitare le ali.
6. Hovering: la fase in cui il corpo è mantenuto in equilibrio.
7. Landing: la fase di atterraggio.

Nello specifico si è riprodotto un modello di ciclo alare detto volo battente, partendo dal decollo fino al ciclo continuo quando il gabbiano è in volo. In figura 3.23 sono riportati alcuni frame che ritraggono il decollo di Gull insieme

Figura 3.23: Alcuni frame del decollo di Gull.

a Pinnie. L'azione di questa fase si svolge prevalentemente in verticale in quanto è il momento in cui il gabbiano deve guadagnare slancio. Lo sforzo necessario per superare l'inerzia e guadagnare movimento è superiore all'energia che occorre per mantenere lo slancio che avanza. Durante tale animazione si è tenuto in considerazione anche il peso dato dalla presenza di Pinnie sul dorso del gabbiano. Inizialmente il gabbiano ha bisogno di esercitare un grande sforzo per prendere aria.

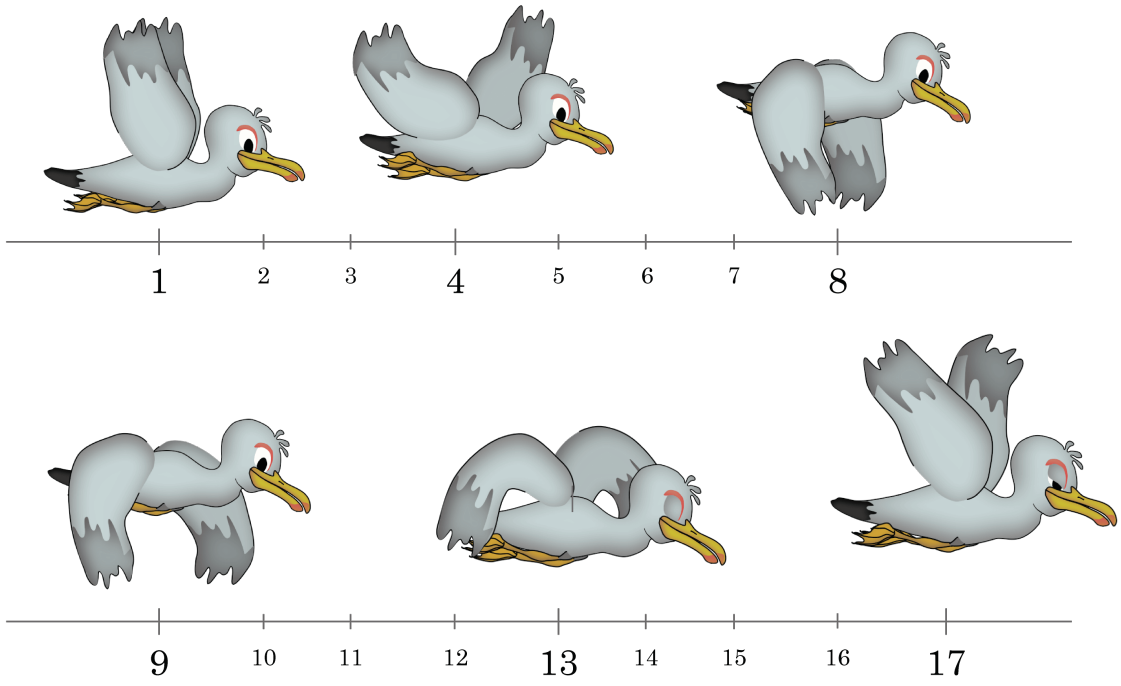


Figura 3.24: I numeri riportati in figura rappresentano i frame presenti nella timeline di Moho. Nello specifico, i numeri evidenziati rappresentano i keyframe e corrispondono alle pose illustrate.

Pertanto è stato illustrato il corpo del gabbiano che si abbassa accovacciandosi rapidamente all'indietro con le ali ravvicinate al corpo, come un'azione che anticipa lo slancio. Successivamente egli incurva le ali verso l'esterno del corpo. Si ha poi un ulteriore piegamento sulle zampe con il corpo slanciato in avanti e le ali distese in verticale. Il gabbiano ora non tocca più il terreno e grazie al curvamento delle ali verso il basso e le zampe completamente distese, si dà la spinta massima per proseguire il volo verso l'alto. Infine si ha la posa del gabbiano con le ali alla massima distensione ed il corpo inclinato in avanti per sfruttare la massima energia della spinta verso il basso. Questa è la posa che precede l'inizio del ciclo alare illustrato in figura 3.24.
All'inizio di tale ciclo, le zampe del gabbiano sono portate all'indietro, verso il corpo, e le ali completamente distese in verticale verso l'alto con le estremità quasi fino a toccarsi l'una con l'altra in modo da ridurre la resistenza dell'aria e per prepararsi

per la successiva posizione. Tutto l'andamento ora si sviluppa principalmente in orizzontale.

Generalmente la parte interna delle ali compie dei movimenti ascendenti e discendenti, mentre la parte più esterna delle api si distende e si ritrae per tutta la durata del movimento. All'estremità superiore e a quella inferiore dei battiti delle ali si ha un movimento più lento rispetto ad altri punti. Si ha una notevole resistenza sulla punta delle ali che di conseguenza portano ad una graduale sovrapposizione dell'azione. Quando le ali si trovano verso il basso, le giunture in prossimità delle spalle possono ancora muoversi verso l'alto. Le punte delle ali iniziano a muoversi verso il basso solamente dopo che la forza della spalla raggiunge le punte delle ali.

Parte II

Dimensione fisica

Capitolo 4

Il cubo: storie di interfacce tangibili

— Elisa Musolino, Stelian Spulber

4.1 Sul cubo

In aggiunta al filmato di animazione delle cui fasi si è parlato nei capitoli precedenti, una seconda parte del progetto riguarda la progettazione di un dispositivo di gioco dotato di display, sensori e una serie di componenti elettronici.

L'idea iniziale alla base di questo dispositivo è stata quella di poter approfondire le tematiche affrontate nel cortometraggio attraverso un oggetto fisico, tangibile e manipolabile che consentisse sia di essere utilizzato autonomamente sia in combinazione, interagendo, con il cortometraggio.
Mentre con il filmato di animazione si è voluto proporre una riflessione sul complesso tema dei cambiamenti climatici, in questa seconda parte l'intento è stato quello di dare la possibilità di scoprire cosa il singolo individuo può fare e mettere in evidenza lo spirito collaborativo necessario per affrontare tematiche complicate come quella trattata. La dimensione sociale infatti, manifestata sotto forma di collaborazione, discussione e interazione tra utenti, è uno degli aspetti chiave che si è voluto approfondire.

Il prototipo del nostro oggetto manipolabile è nato da una serie di considerazioni e caratteristiche:

1. Innanzitutto la facilità di manipolazione e l'aspetto minimale, al fine di consentire a chiunque di potervi interagire in modo intuitivo e naturale.

2. Questo avrebbe dovuto essere dotato di un display touchscreen pertanto sono state escluse alcune forme più *esotiche*.

3. Ultimo concetto chiave riguarda la modularità del sistema che avrebbe dovuto permettere l'utilizzo di più dispositivi dello stesso tipo, combinandoli l'uno con l'altro al fine di creare un'esperienza che coinvolgesse gli utenti sotto diversi punti di vista.

Per rispondere all'ultima caratteristica di cui il sistema doveva essere dotato è necessario introdurre il concetto di *space tessellation*, cioè un riempimento dell'intero spazio infinito, realizzato affiancando solidi geometrici in modo da non lasciare spazi vuoti. Una tassellazione viene detta regolare se i suoi tasselli sono solidi tutti congruenti l'uno con l'altro e sono disposti in modo che due solidi abbiano in comune al più un vertice, uno spigolo o una faccia, oppure spigoli o facce consecutive [1]. Un esempio di tassellazione regolare è rappresentato nella figura 4.1.

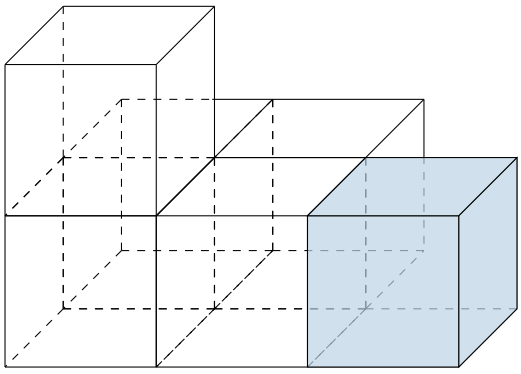


Figura 4.1: Tassellazione regolare dello spazio con il cubo.

Nel *Timeo*, scritto di Platone, uno dei passaggi più affascinanti si ha nella descrizione dei cinque poliedri regolari con cui Platone stabilisce una relazione fra essi e gli elementi fondamentali della composizione dell'universo[2]. Questa parte del dialogo di Socrate con il pitagorico Timeo, unisce l'immaginazione poetica con il rigore della matematica e della geometria.
Il cubo è uno dei cinque poliedri regolari, o *corpi platonici*, indicati da Platone in [3] come "bellissimi corpi, fra di loro dissimili, di cui alcuni possono, dissolvendosi, generarsi reciprocamente". Le figure piane costitutive di questi poliedri sono tre poligoni regolari: il triangolo equilatero, il quadrato e il pentagono. Per Platone il cubo corrisponde a uno dei quattro elementi costitutivi dell'universo: la terra; egli scrive di aver assegnato la terra alla figura cubica, in quanto fra i quattro elementi, è quella meno soggetta al movimento e quindi quella che necessita della maggiore solidità.

La forma geometrica che ci è sembrata più adatta alle nostre intenzioni, al di là delle considerazioni *platoniche* prima esposte, è quella del cubo. Questo esaedro infatti è l'unico tra i cinque poliedri regolari, che con le sue repliche è in grado di riempire lo spazio con regolarità, figura 4.1, permettendo di concretizzare il concetto di modularità in uno spazio tridimensionale. Si possono quindi usare i cubi affiancandoli l'uno con l'altro su un tavolo, ma si possono anche impilare, aumentando in questo modo lo spazio creativo degli utenti.

Per la fase di prototipazione del sistema CuBee sono stati realizzati un totale di cinque cubi, come il numero dei corpi platonici. Questo numero non è fisso, ma è stato reputato sufficiente a dimostrare le potenzialità del sistema in quanto permette un buon grado di componibilità tra i cubi al fine di realizzare un'esperienza d'uso sostanziosa per l'utente. Allo stesso tempo è stato un compromesso tra i costi non indifferenti per la realizzazione fisica degli oggetti e la funzionalità del sistema nelle sue articolazioni.

(a) Brightstarts **(b)** Janod Blocks

(c) Plan toys **(d)** Teepao Mattoncini

Figura 4.2: Esempi di giochi per bambini che richiamano la forma del cubo. Fonte: Amazon.

Inoltre il cubo è anche uno dei primi oggetti tridimensionali con cui i bambini hanno a che fare fin dai primi mesi di vita, e sul mercato sono presenti innumerevoli giochi dalle fattezze cubiche e pensati per i bambini nella fascia 0-3. Questi giochi, di cui in figura 4.2 ne sono mostrati alcuni, stimolano i bambini a sviluppare coordinazione e spirito di osservazione, favoriscono l'immaginazione e la memoria visiva e attraverso

diverse texture, suoni e colori contribuiscono allo sviluppo del tatto, dell'udito e della percezione.

Essendo quindi il cubo estremamente diffuso nel nostro contesto socio-culturale ciò contribuisce anche alla familiarità degli utenti con il dispositivo che ci siamo proposti di sviluppare.

4.2 Case studies

Il cubo ricorre in numerosi giochi, controller e dispositivi. Quello di Rubik è probabilmente il gioco più famoso in questo senso, ma non mancano altri prodotti commerciali affini alla forma del cubo.

Nei successivi sottoparagrafi verranno analizzati due dei dispositivi che si avvicinano maggiormente all'idea che è alla base di questo progetto: entrambi sono caratterizzati da una complessità progettuale notevole ed è interessante confrontare le differenti modalità di funzionamento e le scelte effettuate dai rispettivi sviluppatori.

4.2.1 Sifteo

I *Sifteo Cubes*, mostrati in figura 4.3, hanno rappresentato l'evoluzione dei *Siftables*, un prototipo sviluppato da David Merrill e Jeevan Kalanithi al MIT Media Lab e reso pubblico nel febbraio 2009 nel corso di una TED Conference[4].

Figura 4.3: Sifteo Cubes. Fonte: Sifteo.

Sono stati presentati sul mercato nel corso dell'edizione 2011 del CES di Las Vegas

e sono rimasti in produzione fino al 2014, quando Sifteo Inc. è stata acquisita da 3D Robotics e il progetto Sifteo Cubes è stato abbandonato [5].

I Sifteo Cubes vengono descritti come "miniature computers that interact with each other, sense their motion, and display rich graphic" [6] e come dispositivi che trasformano il "2-D gaming into 3-D motion-aware blocks that can be tilted, jostled, rotated, and clicked to play a variety of games" [7]. Il target primario di questi device era costituito da bambini dai 6 ai 12 anni.

I *cubi*, dalle dimensioni di $38 \times 38 \times 14$ mm, contenevano al proprio interno:

- un processore ARM a 32 bit;
- un display LCD TFT touchscreen da $1''$ con risoluzione 128×128 px;
- la tecnologia proprietaria di Near Field Object Sensing;
- un accelerometro a tre assi;
- una batteria ricaricabile che garantiva fino a 4 ore di vita.

I prototipi Siftables usavano lo stesso microcontroller ATmega644 della console Uzebox e il loro costo di produzione ammontava a circa 200 dollari l'uno [8]. La maggior parte dei componenti presenti nei prototipi Siftables provenivano da Sparkfun, negozio online con un vasto assortimento di componenti elettronici a prezzi accessibili. I cubi disponevano della capacità di:

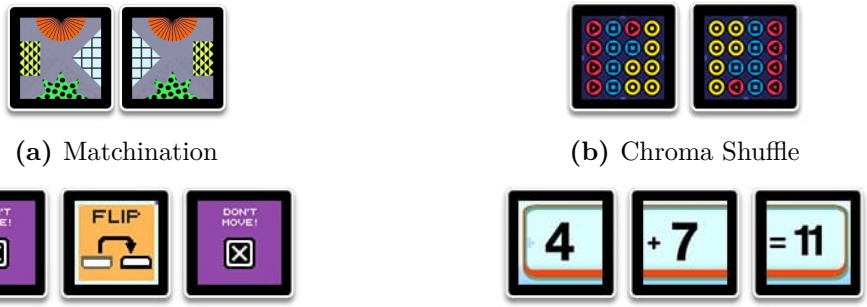
- comunicare l'uno con l'altro in modalità wireless;
- riconoscere i cubi vicini e quale faccia è appoggiata su quale faccia;
- rilevare la propria inclinazione.

Queste semplici idee, combinate l'una con l'altra, si prestavano a una grande varietà di giochi. Alcuni di questi, resi successivamente disponibili per i Sifteo, sono:

- *Matchination*: in questo gioco si devono combinare le facce dei cubi, sui quali sono rappresentate delle forme con un riempimento diverso, e che sono accomunate da caratteristiche quali la forma, il colore o il pattern. Nella modalità di gioco Easy, si devono combinare le forme affini nel minor tempo possibile, mentre nella modalità Arcade, le facce si riempiono progressivamente e le forme devono essere eliminate connettendo i cubi giusti al fine di non riempire lo schermo del cubo. Figura 4.4a.
- *Chroma Shuffle*: in questo gioco si parte da una griglia di puntini colorati che occupano tutto lo schermo. L'obiettivo è quello di far sparire i puntini; per fare ciò si devono trovare due cubi che hanno i puntini dello stesso colore e connettere i puntini uguali. Si vince solamente se tutti i puntini sono spariti. Se ad esempio rimane un solo puntino blu in un cubo e due puntini verde e

rosso negli altri, non si può vincere. Per riempire nuovamente un cubo lo si può capovolgere. Figura 4.4b.

- *Do the Sift*: per iniziare a giocare bisogna allineare i cubi uno in fila all'altro; su ogni cubo vengono indicate man mano delle azioni da svolgere in un tempo predefinito, ad esempio può essere richiesto di capovolgere il cubo, di ruotarlo o di non muoverlo affatto. Si perde al raggiungimento di tre strike, ovvero se per tre volte non si compie l'azione giusta o ci si impiega troppo tempo. Al contrario, si vince se si completano almeno 40 azioni. Figura 4.4c.
- *Cube Math*: in questo caso si posizionano vicini due o più cubi con il numero sul display. Quando questi vengono affiancati si accenderà il segno della sottrazione o della somma, mentre su un altro cubo comparirà il segno dell'uguaglianza. Per cambiare l'operazione matematica si può cliccare sul display dei cubi, mentre per cambiare numero è possibile scuoterli. Figura 4.4d.



(a) Matchination **(b)** Chroma Shuffle

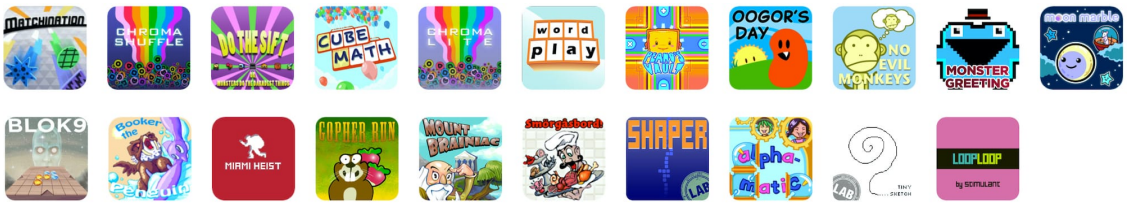
(c) Do The Sift **(d)** Cube Math

Figura 4.4: Alcuni giochi presenti sui Sifteo Cubes. Fonte: Sifteo.

In figura 4.5 sono presenti i giochi disponibili su SiftRunner, la libreria dei Sifteo, al 10 settembre 2015 [9]. Nel corso degli anni l'assortimento dei giochi disponibili non ha avuto l'aumento sperato, il che, unito al costo dei giochi compreso tra gli 8 ed i 12 dollari, probabilmente non ne ha decretato la diffusione e il successo sperati. Il *Creativity Kit*, uno strumento per far creare agli utenti dei giochi in cui si devono ordinare elementi diversi come lettere, parole o numeri, è stato uno degli strumenti più apprezzati e recensiti.

Nei tre anni di produzione sono state lanciate due generazioni del prodotto. La prima è stata resa disponibile nel settembre 2011 [6]. Poco più di un anno dopo è arrivata la seconda, e ultima, generazione. Entrambe le generazioni dei Sifteo Cubes prevedevano l'utilizzo contemporaneo di un massimo di sei cubi.

Per quanto riguarda la prima generazione, era necessario il lancio di un applicativo su un computer, il quale forniva la potenza di calcolo necessaria per la generazione grafica dei contenuti dei singoli dispositivi. La comunicazione, wireless, tra cubi



(a) Giochi della 1° generazione

(b) Giochi della 2° generazione

Figura 4.5: Il catalogo dei giochi presenti su SiftRunner. Fonte: Sifteo.

e computer avveniva tramite l'ausilio di un dongle USB incluso nella confezione, che copriva una distanza di circa 3 metri. Il computer quindi era necessario non soltanto per caricare uno dei giochi della libreria Sifteo ma da esso vi dipendevano i cubi; quindi in assenza non era possibile utilizzarli in alcun modo.

Il requisito del computer che forniva non solo la potenza di calcolo richiesta dai dispositivi, ma anche la musica e gli effetti sonori dei giochi, è stato uno degli aspetti più negativi del sistema: si pregiudicava in questo modo l'aspetto portabile di cui questi dispositivi disponevano intrinsecamente, limitandone l'uso nelle strette vicinanze di un computer e rendendoli vincolati all'ambito domestico. Questo aspetto è stato solo in parte migliorato con l'arrivo della seconda generazione, che non richiedeva più l'intermediazione di un computer ma usava un'unità centrale, figura 4.6, che ne prendeva in carico le funzioni.
Ad ogni modo un altro ostacolo alla diffusione e al successo commerciale di questi dispositivi è stato quello del prezzo: 149 dollari per tre cubi il cui ecosistema, quanto meno inizialmente, disponeva di poco più di una decina di giochi e non tutti gratuiti, è stato reputato troppo alto, sebbene il costo è sceso a 129 dollari, per lo stesso set, con la seconda generazione.

Il periodo di vita di questi dispositivi è stato inoltre influenzato dalla contemporanea e sempre maggiore diffusione di dispositivi come l'iPhone e l'iPod Touch che racchiudevano in un unico device molteplici funzionalità, in combinazione con la costante crescita dei cataloghi di contenuti di intrattenimento, tra cui i principali erano Play Store, App Store ed iTunes Store.

Figura 4.6: La seconda generazione dei Sifteo Cubes. Fonte: Sifteo.

D'altronde il problema si ripropone anche oggi: quale futuro può avere un dispositivo di gioco che abbia un prezzo equiparabile ad uno smartphone di fascia media e che può accedere a milioni di contenuti di intrattenimento?

4.2.2 Radica Cube World

Radica Games Limited è un'azienda di giocattoli elettronici, acquisita nel 2006 da Mattel Inc. Uno dei prodotti sviluppati, ideati da Tory Ellis, è *Cube World*, visibile in figura 4.7: si tratta di un ecosistema costituito da una serie di dispositivi elettronici dotati di uno schermo LCD bianco e nero, i quali racchiudono al proprio interno personaggi stilizzati che compiono determinate azioni e che possono interagire tra di essi connettendo più cubi.

Figura 4.7: Radica Cube World. Fonte: Mattel.

Tra il 2005 e il 2008 ne sono state realizzate cinque diverse versioni, ognuna comprendente quattro dispositivi tematici, identificati ciascuno da un personaggio.

Ogni dispositivo ha l'aspetto di un cubo con lato di 47 mm. La sesta e ultima serie, *Jumbo*, prevedeva due diversi cubi, quattro volte più grandi rispetto ai precedenti.

Il set da due cubi aveva un costo di 19,95 sterline inglesi. Questo gioco era consigliato per bambini al di sopra degli 8 anni.

Le caratteristiche dei Radica Cube World erano le seguenti:

- ogni cubo era provvisto di contatti magnetici sulle quattro facce laterali attraverso i quali poteva rilevare la presenza di altri cubi e il collegamento di un cubo con l'altro;
- ogni cubo disponeva di tre pulsanti attraverso i quali gestire il dispositivo e interagire con il personaggio al suo interno;
- ogni cubo era inoltre dotato di un sensore che permetteva ad esempio di far ruotare il personaggio all'interno del cubo.

Su ciascun cubo vi era riportato il nome del personaggio ed il simbolo dell'oggetto con cui esso interagiva. Ad esempio, per quanto riguarda la prima serie i personaggi sono Slim, Doger, Scoop e Whip che interagiscono rispettivamente con un bastone, una palla, un cane e una corda.
Per ogni personaggio erano previste circa 100 animazioni in cui esso eseguiva delle azioni. I cubi rimanevano attivi per circa 5 minuti, oltre i quali, in assenza di interazione dell'utente, si disattivavano.

L'aspetto più interessante era dato dalle possibili combinazioni dei cubi e le molteplici interazioni che inevitabilmente si venivano a creare: era possibile combinare e usare fino a 16 cubi, mentre i dispositivi della quinta serie potevano essere utilizzati con i cubi della prima serie e viceversa. Tuttavia gli oggetti dei personaggi della prima serie non potevano essere utilizzati dai personaggi delle serie successive e viceversa, ma l'interazione era limitata ai soli personaggi.

Connettendo due o più cubi i personaggi potevano incontrarsi in uno di essi e interagire l'uno con l'altro: se si connettevano quindi Scoop e Doger, il personaggio presente in Doger poteva lanciare la palla al cane di Scoop, il quale la riportava *attraversando* il collegamento creato tramite l'avvicinamento dei due cubi. All'interno di ogni cubo potevano incontrarsi fino ad un massimo di quattro personaggi.

Un altro aspetto degno di nota riguarda la semplicità dei dispositivi nella grafica e nel loro utilizzo, mentre le principali critiche a questi giochi riguardano la ripetitività delle azioni e la mancanza di controllo e interazione con l'utente. La maggior parte delle sequenze di animazione definite per un cubo possono essere visualizzate nell'arco di pochi minuti. Poiché i personaggi possono utilizzare i giocattoli degli altri personaggi, in combinazione con il proprio giocattolo, avere un set completo

da quattro cubi riduce in parte questa ripetitività che in definitiva può portare all'abbandono del gioco.

Figura 4.8: Cube World Serie 2. Fonte: Mattel.

Capitolo 5

Progettazione: dall'idea al prototipo fisico

— Elisa Musolino, Stelian Spulber

La progettazione dei dispositivi di gioco è stata articolata in una serie di fasi, dall'ideazione iniziale del dispositivo, ossia le sue caratteristiche e modalità di funzionamento, alla realizzazione del primo prototipo funzionale, ossia il raggiungimento di un dispositivo che soddisfa la maggior parte dei requisiti stabiliti.

Alcuni aspetti ai quali sono state rivolte le dovute attenzioni, riguardano innanzitutto i bisogni degli utenti: si è cercato di capire se l'idea iniziale avrebbe potuto soddisfare i bisogni degli utenti di riferimento all'interno di un contesto ben definito, quale gli spazi di un museo.
In secondo luogo si è tenuto conto degli aspetti legati alla facilità di utilizzo del dispositivo, il quale, rientrando in un'installazione museale, avrebbe richiesto che ogni sua funzione fosse facilmente intuibile e comprensibile dall'utente, in modo che l'esperienza risultasse fluida, non frustrante e piacevole.
Infine si è cercato di tener conto anche di una particolare problematica che riguarda molte installazioni interattive all'interno dei musei, soprattutto quelli rivolti ai più giovani e con numerose tecnologie interattive, di complessità diversa; queste, infatti, possono generare problemi di stabilità, errori e malfunzionamenti che comportano un disservizio al pubblico e richiedono la presenza di personale qualificato per il ripristino di tali installazioni, rappresentando quindi un costo per il museo. Per questo progetto sono state prese opportune precauzioni dal punto di vista del design dei dispositivi ma anche dal punto di vista hardware e software, con l'obiettivo di realizzare un sistema robusto e affidabile, relativamente autonomo.

Nei successivi paragrafi verrà affrontato il processo di ideazione ed i requisiti che il cubo avrebbe dovuto avere nonché le fasi che hanno trasformato l'idea in un dispositivo tangibile, partendo da uno spazio puramente virtuale; pertanto verrà trattata la fase di modellazione e la successiva stampa dei dispositivi, con le dovute considerazioni relative al progetto. Successivamente, nel capitolo 6, si continuerà la descrizione della progettazione del dispositivo dal punto di vista hardware e, nel capitolo 7, dal punto di vista software.

5.1 Requisiti

In fase di progettazione del sistema sono stati definiti alcuni aspetti riguardanti il funzionamento dei dispositivi di gioco. Innanzitutto i cubi devono potersi connettere l'uno con l'altro, dando forma a combinazioni spaziali che permettono di aumentare l'interesse e la curiosità degli utenti verso i contenuti presentati. Tali dispositivi devono anche disporre di una o più modalità di feedback sulle azioni dell'utente, a seconda della combinazione, del posizionamento o del movimento cui ogni cubo è sottoposto.
Inoltre il sistema CuBee deve essere robusto e devono essere presi opportuni accorgimenti in fase di sviluppo. Questi ultimi sono dati dal contesto museale in cui il sistema è stato inserito, il quale prevede innanzitutto un utilizzo non esclusivo, ossia non limitato all'uso da parte di un singolo utente. Inoltre, l'introduzione al funzionamento del sistema e alle modalità di gioco, è tale da permettere all'utente un rapido apprendimento che ne faciliti l'utilizzo e renda agevole l'esperienza interattiva.

Le caratteristiche principali alla base del funzionamento del sottosistema costituito dai cubi sono le seguenti:

1. Indipendenza: ogni cubo deve contenere al suo interno i componenti necessari per l'input, l'elaborazione e l'output dei dati, senza richiedere l'ausilio di un ulteriore dispositivo che gestisca parte di questi aspetti.

2. Modularità: il numero di cubi utilizzabili è arbitrario e questi possono essere connessi l'uno con l'altro senza vincoli, tenendo in considerazione la presenza del display che esclude lo sviluppo di certe configurazioni spaziali in cui il display non sia rivolto verso l'utente.

3. Multisensorialità: attraverso i cubi si vuole impegnare non solo la vista e l'udito ma anche il tatto, mediante l'uso di un materiale naturale, ruvido e con imperfezioni come il legno.

Il primo di questi tre punti, l'indipendenza dei cubi, comporta che ciascun cubo è dotato dei componenti necessari per gestire autonomamente i dati dei propri sensori e generare la grafica del display e gli altri output come la componente audio e l'illuminazione delle facce laterali. Ciò ha comportato un costo rilevante sia in termini economici sia in termini puramente computazionali in quanto un sistema distribuito comporta una maggiore complessità.
Il secondo punto rappresenta invece la caratteristica principale del sistema. Con il termine *modularità* si fa riferimento alla possibilità di utilizzare un numero arbitrario di cubi e alle varie opzioni di composizione consentite. Al fine di applicare tale concetto è necessario che il singolo cubo sia in grado di:

1. Riconoscere la presenza degli altri cubi connessi sulle proprie facce.
2. Identificare per tali cubi quali siano le rispettive facce coinvolte nella connessione.

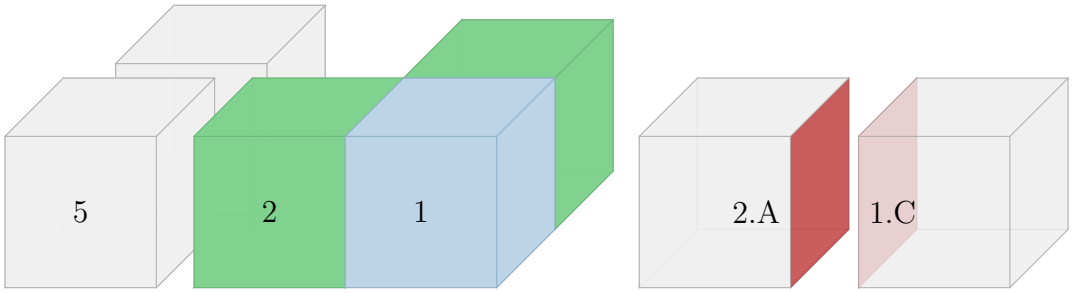


Figura 5.1: (a) Riconoscimento dei cubi connessi e (b) identificazione delle facce coinvolte nella connessione.

Queste due funzionalità sono illustrate nella figura 5.1: il cubo 1, blu, è in grado di riconoscere i cubi connessi sulle proprie facce, identificati con il colore verde nella figura 5.1a, mentre non è in grado di stabilire una connessione con i cubi grigi in quanto distaccati da esso; inoltre, come si nota nella figura 5.1b, il cubo 1 è in grado di identificare ad esempio, nel caso della connessione con il cubo 2, le due facce coinvolte evidenziate in rosso, ossia la propria (1.C) e quella del cubo 2 (2.A).
Per quanto riguarda l'ultimo punto si è scelto di far confluire nei dispositivi non solo una dimensione visiva, data dal display e dall'illuminazione delle facce, ma anche una dimensione sonora, al fine di coinvolgere maggiormente l'utente. Oltre a queste caratteristiche, la peculiarità dei cubi è sottolineata dall'utilizzo di un materiale che risulta ruvido ma gradevole al tatto, dotato di irregolarità e che richiama i giochi dell'infanzia.

Ciascun cubo facente parte del sistema è stato quindi dotato dei componenti elettronici necessari per rispondere alle caratteristiche di cui sopra. In aggiunta alle

funzionalità legate alla modularità, le specifiche dei dispositivi alle quali rispondere tramite la progettazione hardware, affrontata nel capitolo 6, sono le seguenti:

1. Rilevare la rotazione, l'inclinazione e lo scuotimento lungo i tre assi e l'orientamento dei cubi al fine di ricostruirne la posizione nello spazio e derivarne il verso del display.
2. Implementare una modalità di comunicazione con gli altri cubi connessi che permetta di scambiare i dati necessari alla generazione della grafica e relativi alle dinamiche di gioco.
3. Dare un feedback visivo all'utente attraverso l'illuminazione delle facce laterali.
4. Offrire una dimensione sonora in cui far immergere l'utente, con l'obiettivo di rendere più coinvolgente l'esperienza d'uso.

Implementare queste funzioni nel prototipo finale del dispositivo di gioco ha richiesto lo studio di diverse soluzioni possibili, con vantaggi e soprattutto limiti annessi. Questi ultimi, intervenuti nella fase di progettazione, sono scaturiti dalle dimensioni del cubo e dei componenti utilizzati, dal formato rettangolare del display e dalle esigenze di posizionamento di alcuni componenti quali ad esempio i LED per l'illuminazione delle facce ed i sensori di prossimità.

Tra gli aspetti menzionati precedentemente, quelli che hanno posto i vincoli più significativi nello sviluppo del progetto e con le ripercussioni più importanti sul prototipo finale, sono indubbiamente il display e le funzionalità relative alla modularità del sistema.

5.2 Prototipazione

Dal punto di vista puramente tecnico il prototipo rappresenta il primo esemplare di una serie di manufatti. In [1] un prototipo viene definito in senso più ampio come "an approximation of the product along one or more dimensions of interest", facendo rientrare in questa definizione qualunque "entità" che presenta almeno un aspetto rilevante per il team di progettazione, per cui vengono inclusi non solo i prototipi fisici, che hanno lo scopo di approssimare un prodotto e che interessano ai fini di questo progetto, ma anche i prototipi analitici come le simulazioni al computer, i modelli analitici, i sistemi di equazioni che descrivono un prodotto ecc.
Dal punto di vista dell'utilizzo, un prototipo viene usato sia nel contesto del marketing per verificare l'interesse di un gruppo di potenziali clienti sia nel contesto della progettazione di un prodotto in cui si rende utile soprattutto per verifiche di tipo funzionale.

Lo sviluppo di un generico prodotto si articola in una serie di fasi sequenziali ma allo stesso tempo interconnesse, che permettono di ottimizzare le risorse messe in campo da una società per arrivare infine ad un prodotto che risponda ai bisogni degli utenti di riferimento. In questo processo di sviluppo la fase di prototipazione ricopre un ruolo di primaria importanza: è solo tramite i diversi prototipi che si susseguono nel periodo precedente alla produzione in serie che è possibile intervenire tempestivamente sul prodotto, riducendo tempi e costi della produzione altrimenti significativi.

Andrea Gatto e Luca Iuliano in [2] individuano nella cosiddetta prototipazione rapida quella "tecnologia innovativa che rende possibile la produzione, in poche ore e senza l'uso di utensili, di oggetti di geometria comunque complessa, direttamente dal modello matematico dell'oggetto realizzato su di un sistema CAD tridimensionale". Tale definizione si ritrova anche in [3], dove si parla più propriamente di un insieme di processi produttivi "capable of producing complex freeform solid objects directly from a computer model of an object without part-specific tooling or knowledge".
In letteratura la prototipazione rapida viene anche indicata come Solid Freeform Fabrication (SFF) ed è diventata di rilievo negli ultimi vent'anni sebbene abbia preso vita a metà degli anni '80 con la stereolitografia. Al pari di analoghe tecnologie sviluppatesi nel corso degli anni, la stereolitografia consiste essenzialmente nella realizzazione di oggetti fisici a partire da modelli CAD tridimensionali.

Un ulteriore strumento di ausilio alla prototipazione, e più in generale alla progettazione di un prodotto, è dato dal CAD 3D. La modellazione tridimensionale consiste nel processo di definizione di una forma tridimensionale all'interno di uno spazio virtuale, il tutto generato attraverso un computer e per tale motivo si parla di 3D Computer-Aided Design. I modelli 3D vengono realizzati mediante opportuni software di modellazione quali 3DS Max, Fusion 360, Blender ecc. I vantaggi della modellazione tridimensionale sono dati dalla facilità di visualizzazione di un determinato aspetto di un prodotto, la possibilità di creare immagini fotorealistiche per la valutazione delle caratteristiche estetiche del prodotto, la possibilità di effettuare analisi e simulazioni relative alle caratteristiche fisiche del prodotto ecc. Interessante è soprattutto quest'ultima potenzialità offerta dalla Computer-Aided Engineering (CAE), rappresentata da tutti quegli strumenti software (MATLAB, Mathematica, Simulink ecc.) che attraverso il calcolo numerico permettono la risoluzione di aspetti legati alla distribuzione delle sollecitazioni di un sistema, al movimento cinematico e dinamico di meccanismi anche complessi ecc.

La prototipazione classica prevede l'utilizzo di tecnologie complesse quali le macchine a controllo numerico (CNC) e richiedono tempi molto più lunghi rispetto a quanto ottenibile con la SFF, la quale permette in poche ore di passare da una concezione virtuale ad un oggetto con una consistenza fisica.

Le tecnologie rientranti nella Solid Freeform Fabrication si basano su un approccio di produzione a strati, contrapposto a metodi di produzione quali lo stampaggio, la fresatura, la tornitura ecc. La rappresentazione di una figura geometrica solida viene decomposta in tante sezioni di spessore infinitesimo, tramite un processo denominato *slicing*, ed un oggetto viene realizzato mediante la progressiva aggiunta di materiale, uno strato dopo l'altro.

La prototipazione quindi è un potente strumento a disposizione del progettista e permette di ottenere molteplici vantaggi strategici, abbassando innanzitutto i tempi e i costi di produzione dei prototipi, rendendo possibile l'eliminazione di eventuali errori di progettazione del modello CAD e permettendo la valutazione delle funzionalità del prodotto. Inoltre, il crescente numero di materiali disponibili consente di ottenere risultati sempre più simili a quelli del prodotto finale, permettendo quindi anche valutazioni dal punto di vista del design.

I prototipi e l'evoluzione del dispositivo di gioco del sistema CuBee verranno trattati in dettaglio nel paragrafo 5.5.

5.3 CAD 3D

Con il termine CAD 3D si intende l'insieme di tutti quei processi di definizione di una forma tridimensionale all'interno di uno spazio virtuale svolti con l'ausilio di un computer. I modelli 3D nati da tali processi sono quindi una rappresentazione digitale di un oggetto, qualunque esso sia.
L'approccio tridimensionale nel CAD è riconducibile ad una logica di *integrazione* che prevede di tener conto già in fase di ideazione del prodotto sia degli aspetti estetici e funzionali sia di quelli produttivi.

La modellazione tridimensionale può avere diversi impieghi e il primo di questi, nonché quello che pone maggior interesse per questo progetto, è dato dall'utilizzo della modellazione per la realizzazione fisica dell'oggetto mediante un opportuno processo di produzione, che in questo progetto è la stampa 3D. La creazione del modello ha quindi come obiettivo quello di trasportare il concetto virtuale nel mondo reale. A tal fine, è richiesto un alto livello di accuratezza, soprattutto per quanto riguarda oggetti composti da più parti, le quali devono essere opportunamente dimensionate per dare forma all'oggetto.
Un altro utilizzo della modellazione tridimensionale è quello che ha come scopo la generazione di rendering, ossia immagini generate al computer che possono essere impiegate per l'illustrazione (*image-making*) o per l'animazione. In questo caso non è sufficiente il solo modello 3D ma si rende necessario definire un punto di vista sulla scena (camera) e una o più sorgenti di luce. La necessità in questo ambito

non è tanto quella di disporre di modelli accurati ma occorre che tali modelli siano piacevoli e convincenti dal punto di vista estetico e diano un senso di realismo all'immagine, se il caso.

I due impieghi sono chiaramente legati l'uno all'altro in quanto entrambi richiedono la modellazione tridimensionale di un oggetto, seppur con le dovute differenze di concezione. Questo paragrafo si concentrerà soprattutto sull'aspetto della modellazione relativamente alla realizzazione fisica di un oggetto.

Modelli 3D

Un modello 3D è definito da una serie di informazioni legate alla propria geometria e dalle caratteristiche legate alla sua apparenza. La geometria di un oggetto rappresentato in uno spazio virtuale è definita da una serie di entità geometriche basilari, definite primitive. Alcune primitive geometriche possono essere descritte da semplici equazioni: in 2D si ritrovano le linee, archi di ellisse, rettangoli e curve polinomiali di basso ordine, mentre in 3D le primitive sono rappresentate da cubi, sfere, cilindri e altre superfici polinomiali di basso ordine[4]. Tali primitive semplici, nel caso bidimensionale possono essere rappresentate tramite equazioni in forma implicita, $f(x, y) = 0$, in forma esplicita, $y = f(x)$ o in forma parametrica, con $x = x(u)$ e $y = y(u)$, dove u è un parametro. Di queste rappresentazioni la forma più adatta per il CAD è data dalla forma parametrica, come spiegato in seguito.

In alcuni ambiti applicativi come nel caso dell'animazione e dell'image-making non interessano i volumi degli oggetti, ma unicamente le superfici. In questo caso quindi, viene definita solamente la geometria del modello ma non vi è alcuna informazione riguardante la topologia dell'oggetto; motivo per cui se due superfici hanno uno spigolo in comune, non viene conservata la relazione tra di esse. In ambito CAD la modellazione attraverso solidi è invece fondamentale per la produzione industriale di un oggetto e viene sempre più utilizzata anche per verifiche di tipo strutturale e per simulazioni.

Modellazione solida parametrica

La modellazione solida, nello specifico la CSG (Constructive Solid Geometry), permette di realizzare oggetti geometrici complessi a partire da una serie di forme elementari come il cubo, la sfera, il cilindro ecc. La costruzione avviene gradualmente attraverso tre possibili operazioni booleane elementari: unione, differenza e intersezione; inoltre, le operazioni CSG possono essere descritte tramite un albero, figura 5.2, in cui ciascun nodo contiene una delle tre operazioni booleane appena citate mentre le foglie sono costituite dalle primitive utilizzate. Questo rappresenta

il primo approccio possibile per la creazione di un oggetto tramite software CAD 3D.

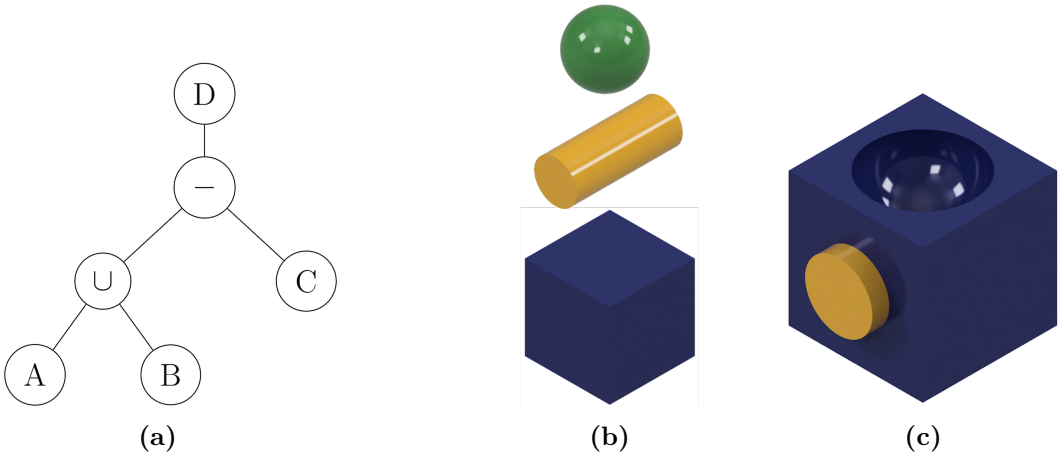


Figura 5.2: Albero delle costruzioni (a) utilizzato con le primitive illustrate in (b) per ottenere il modello rappresentato in (c).

Complessivamente la modellazione basata su superfici e la modellazione solida possono creare oggetti esattamente identici dal punto di vista funzionale; le differenze sono date dal modo in cui essi vengono creati, modificati e dalle approssimazioni che vengono introdotte, come nel caso della modellazione poligonale. In quest'ultima forma di rappresentazione infatti, le superfici sono costituite da un numero più o meno elevato di poligoni, congiunti attraverso i bordi o i vertici; la risoluzione, che comporta un utilizzo maggiore di memoria e dei tempi di rendering qualora tale operazione si rendesse necessaria, rappresenta in definitiva il discriminante tra le due modalità di modellazione citate.

La modellazione solida parametrica si contrappone a quella solida classica, in cui ad esempio una volta assegnate le dimensioni di un oggetto, considerate costanti e non variabili, si rendeva molto difficile effettuare operazioni di editing per tali parametri e si preferiva ricominciare da capo la modellazione. Ad oggi la modellazione solida parametrica è implementata nella maggior parte dei software CAD e, grazie al concetto delle *feature*, rende del tutto modificabili i parametri assegnati all'oggetto, permettendo anche di impostare relazioni e vincoli di tipo geometrico, matematico o fisico.

In aggiunta alla modellazione solida parametrica i software CAD implementano anche ulteriori tecniche di modellazione, quale ad esempio la modellazione basata su superfici, permettendo in questo modo di realizzare modelli tridimensionali estremamente complessi.

Oltre all'approccio CSG accennato prima, basato sull'applicazione di una serie di operazioni booleane a forme solide elementari, vi sono altri due approcci possibili

per la creazione di un modello solido. Il primo di questi si basa sulle feature e non differisce di molto dal CSG in quanto le primitive solide vengono sostituite con le feature, le quali possono anche contenere operazioni booleane.
Dal punto di vista pratico per realizzare ad esempio un cubo dotato di un foro, usando l'approccio basato sulle primitive i passi sono i seguenti:

1. Creare un parallelepipedo con lati uguali.
2. Creare un cilindro.
3. Eseguire l'operazione di sottrazione del cilindro dal cubo.

Per lo stesso oggetto, l'approccio basato su feature prevede i seguenti passi:

1. Creare un parallelepipedo con lati uguali utilizzando l'apposita feature.
2. Praticare un foro all'interno del cubo attraverso l'apposita feature.

Vi è infine un ultimo approccio per creare un oggetto e si basa sugli sketch, ossia disegni tecnici bidimensionali definiti su di un piano, a partire dai quali è possibile generare forme più complesse. Questo metodo prevede la selezione di un piano, il disegno tecnico attraverso linee, rettangoli, ellissi ecc. e opportune operazioni messe a disposizione dal software CAD, l'assegnazione di eventuali vincoli e l'impostazione di relazioni tra le entità componenti il disegno; l'ultimo passo consiste nel trasporre tali sketch nel mondo tridimensionale, attraverso operazioni come l'estrusione.

All'interno di un ambiente CAD 3D ogni parametro numerico riguardante gli sketch e le feature può essere modificato in qualsiasi fase della modellazione. Il flusso di lavoro quindi non è unidirezionale, come accade per la modellazione poligonale, ma è reversibile e la maggior parte delle operazioni eseguite sul modello possono essere facilmente modificate.

Per quanto riguarda i prototipi del cubo realizzati per questo progetto di tesi, si è inizialmente utilizzato il software di modellazione Rhinoceros, successivamente sostituito da Autodesk Fusion 360. I motivi di tale cambiamento sono stati dettati dalla necessità di ottimizzare il flusso di lavoro, reso più agevole con Fusion 360 grazie al suo funzionamento basato su cloud che ha permesso anche di sfruttare la collaborazione in real-time al progetto di modellazione; ma anche dalla necessità di utilizzare alcune funzionalità tipiche dell'ambiente CAD 3D, quale la modellazione solida parametrica, che ha reso possibile la riduzione dei tempi richiesti per le numerose modifiche intervenute sul modello del dispositivo progettato.

Nel sottoparagrafo successivo verranno affrontati alcuni aspetti del software Fusion 360 e ne verrà illustrato l'uso in relazione al sistema CuBee.

5.3.1 Autodesk Fusion 360

Fusion 360 è il software utilizzato per la modellazione del dispositivo di gioco progettato. Esso riprende alcuni aspetti che hanno permesso il successo di Autodesk in ambito CAD, rendendoli più accessibili, e coniuga queste caratteristiche con nuove funzionalità dedicate alla prototipazione rapida, permettendo la creazione di modelli estremamente elaborati ad un prezzo più accessibile rispetto a software professionali analoghi. Inoltre Autodesk rilascia gratuitamente tale software per uso personale e per uso commerciale in caso di aziende con introiti contenuti.

Fusion 360 utilizza modelli solidi, ossia dotati di volume e massa variabili a seconda del materiale assegnato. I punti di forza del programma sono la possibilità di realizzare modelli estremamente elaborati e la varietà di simulazioni disponibili.

Dal punto di vista delle modalità di lavoro, il programma dispone di numerose funzionalità utili soprattutto in ambito meccanico e industriale, il tutto reso disponibile attraverso un'interfaccia molto più semplificata rispetto ai classici strumenti CAD, permettendo così di raggiungere una sufficiente operatività in un tempo relativamente inferiore.

Per quanto riguarda il design, Fusion 360 offre un ricco set di strumenti ad ausilio della progettazione di un prodotto, permettendo l'uso di diverse modalità di modellazione e strumenti che rendono tale processo alquanto flessibile:

1. Sketching: è possibile creare e modificare sketch inserendo vincoli e relazioni tra le entità geometriche utilizzate; in questa modalità sono presenti numerosi strumenti per il disegno tecnico ed è possibile utilizzare spline e curve complesse. A partire dagli sketch è poi possibile generare e/o modificare modelli 3D attraverso l'operazione di estrusione.
2. Freeform Modeling: permette di creare subdivision surfaces con le T-splines in modo da ottenere superfici lisce senza l'introduzione di approssimazioni come nel caso della modellazione poligonale. La modifica delle superfici, al pari di altre operazioni analoghe, si effettua facilmente mediante un comando intuitivo *Press/Pull*.
3. Surface Modeling: consente di creare e modificare superfici parametriche complesse per disegnare o riparare una determinata geometria del modello. Può essere considerata l'analogo digitale della lavorazione con l'argilla e offre molta libertà creativa all'utente.
4. Parametric Modeling: permette di generare un modello 3D in modo parametrico, come descritto nel paragrafo precedente.

5. Mesh Modeling: è possibile modificare ed eseguire operazioni di *riparazione* di eventuali modelli tridimensionali scannerizzati o modelli mesh generati ad esempio in Blender o con analoghi software di modellazione.

In aggiunta a queste modalità vi è anche la possibilità di eseguire animazioni dei componenti e studiarne il montaggio in modo da verificare la compatibilità delle parti che compongono un determinato prodotto. Fusion 360 fornisce inoltre una serie di facilitazioni per la produzione di un oggetto, in modo da ridurre l'impatto che le modifiche di progettazione possono avere sul processo di progettazione; ad esempio è possibile visualizzare un'anteprima dell'operazione di slicing necessaria per la stampa 3D e valutare diverse opzioni per la generazione dei supporti che permettono la buona riuscita della stampa. Infine si possono generare render del modello realizzato, sfruttando l'opzione di rendering in cloud che velocizza tale operazione.

Uno dei vantaggi di Fusion 360, al pari di altri software CAD, è quello di poter modificare i parametri dei modelli creati in precedenza e propagare tali cambiamenti nel flusso di creazione. Ad esempio per quanto riguarda gli esagoni presenti sulle facce laterali dei cubi, è possibile, tramite una timeline, tornare al punto in cui tale feature è stata creata, modificare la lunghezza del lato dell'esagono e il software eseguirà nuovamente tutti i calcoli e le operazioni tenendo in considerazione la modifica effettuata e le feature successive a tale punto nella timeline. Tuttavia alcune feature necessitano di un piano o di un asse di riferimento e qualora vengano impostati in modo errato questi riferimenti, alcuni cambiamenti dei parametri non vengono correttamente propagati nello storico delle operazioni, causando errori sulle feature successive. Tale funzionalità, denominata Design History, è mostrata in un esempio in figura 5.3.

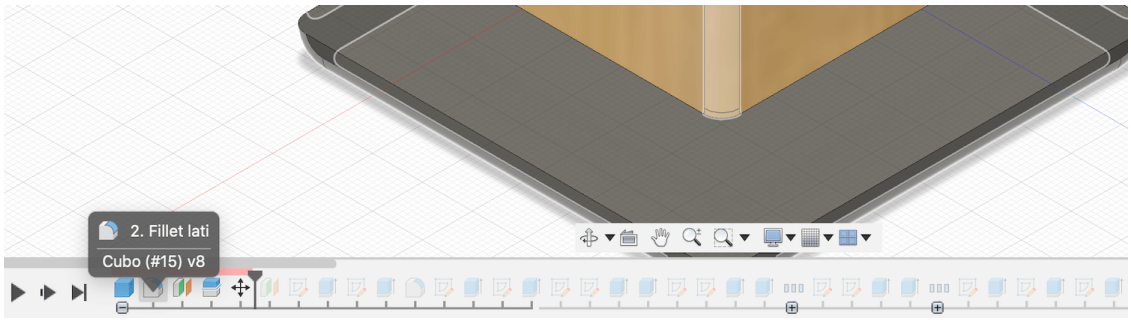


Figura 5.3: Esempio delle feature memorizzate nel Design History del guscio del dispositivo.

Infine, attraverso la definizione di vincoli e relazioni, si possono facilmente modificare aspetti più complessi che altrimenti richiederebbero un numero elevato di operazioni sul modello.

All'interno del software è possibile effettuare anche una serie di simulazioni che possono rendere più fluido il processo di progettazione di un prodotto, e sono le seguenti:

- Static Stress: analizza l'eventuale deformazione e lo stress causato da carichi strutturali e vincoli sul modello.
- Modal Frequencies: determina le frequenze modali del modello.
- Thermal: determina la reazione del modello a carichi di calore, permettendo di visualizzare la distribuzione della temperatura nel modello quando questo viene riscaldato o raffreddato.
- Thermal Stress: determina le temperature e le distribuzioni delle sollecitazioni sul modello risultanti da carichi termici e strutturali, permettendo di capire se il modello verrà deformato quando riscaldato o raffreddato in combinazione con carichi fisici ad esso applicati.
- Structural Buckling: permette di rilevare se componenti lunghi e sottili sottoposti a determinati carichi fisici possono collassare.
- Nonlinear Static Stress: determina la reazione del modello quando il carico applicato è abbastanza grande da deformare in modo permanente la struttura del modello.
- Event Simulation: determina il modo in cui il modello reagisce al movimento, alle deformazioni e all'impatto, come nel caso di una caduta a terra.
- Shape Optimization: ottimizza i componenti del modello per renderli leggeri e strutturalmente efficienti in base ai carichi applicati. Permette quindi di rimuovere il materiale non necessario per la solidità del modello.

Tali simulazioni permettono di risparmiare tempo e risorse all'interno del processo di progettazione e quindi la loro presenza in Fusion 360 rappresenta un ulteriore vantaggio, nonché un valore aggiunto di questo software.

Nel corso del 2018 è stata introdotta all'interno del software una nuova funzionalità di progettazione, ossia il *Generative Design*, noto anche come progettazione generativa o design generativo, alla base del quale si ritrova l'ottimizzazione topologica. Tale tecnologia si basa su determinati obiettivi definiti dal progettista, quali dimensioni, peso, materiali, vincoli strutturali ecc., che vengono opportunamente elaborati da una serie di algoritmi che valutano rapidamente migliaia di scelte di progettazione testando molteplici configurazioni che per un progettista sarebbe difficile valutare altrimenti. In questo modo è possibile generare un maggior numero di soluzioni progettuali per un determinato prodotto o componente, nel rispetto dei requisiti del processo di produzione e delle prestazioni necessarie.

Il Generative Design ha un potenziale enorme in ambito automotive o aerospaziale in cui l'ottimizzazione della massa di un determinato componente genera un risparmio considerevole nella produzione e nei consumi di un velivolo o di un autoveicolo.

Per la realizzazione di questo progetto le parti progettate in Fusion 360 sono relative alla struttura, sulla quale sono stati fissati i singoli componenti elettronici descritti nel capitolo 6, e al guscio di forma cubica che contiene la struttura ed i componenti utilizzati.

Design del guscio

Il guscio del dispositivo è stato suddiviso in due parti, in quanto è necessario inserire al suo interno la struttura con i componenti montati. Esso è stato modellato a partire dalla primitiva del cubo, opportunamente *scavato* per ottenere lo spazio necessario alla struttura e successivamente è stato sezionato in due parti, contenitore e coperchio. Si è scelto di sezionare il cubo alla base, dividendolo in due parti di altezza rispettivamente 81 mm e 7 mm, in modo da far percepire all'utente il dispositivo come un blocco uniforme ed omogeneo; queste due parti sono agganciate l'una con l'altra tramite quattro viti di lunghezza pari a 20 mm.

A partire dal solido del cubo, per le operazioni successive sono stati utilizzati gli sketch e il comando Press/Pull. Ad esempio per la realizzazione degli esagoni destinati all'illuminazione delle facce laterali del cubo, questi sono stati disegnati all'interno di uno sketch e successivamente, tramite il comando Press/Pull, le superfici sono state estruse, eliminando parte della faccia laterale del cubo. Queste due operazioni basilari sono state eseguite anche per creare lo spazio di alloggiamento dei tag NFC, del display, dello speaker, dei magneti, del dado e del bullone necessari per la chiusura del guscio. Sul fondo del guscio, visibile in figura 5.4b, è presente il foro per la ricarica del cubo, attraverso una porta MicroUSB, il foro per accedere al pulsante di accensione e spegnimento del dispositivo e infine l'incisione del logo CuBee.

Poiché i cubi sono cinque e in alcune modalità di gioco si è reso necessario avere un cubo iniziale di riferimento si è scelto di distinguere i cubi attraverso la forma della griglia dello speaker. Il cubo principale è quindi evidenziato da una griglia a forma di ape mentre i restanti cubi dispongono di una griglia a forma ovale, figura 5.4a. Nel disegno di entrambe le griglie si è tenuto conto oltre che dell'aspetto estetico e simbolico anche dell'aspetto prettamente funzionale, pertanto tale griglia è stata modellata creando e posizionando dei fori simmetrici per garantire una diffusione omogenea del suono.

(a) Griglia dello speaker **(b)** Base del guscio

Figura 5.4: Dettagli relativi al guscio del dispositivo

Nel predisporre gli spazi necessari per i componenti si è tenuto conto dell'usabilità del dispositivo: ad esempio si è osservato nelle prime fasi di test con gli utenti che la presa in mano dei cubi tende a concentrarsi nella parte superiore del cubo e sugli spigoli; per questo motivo gli spigoli sono stati ulteriormente arrotondati in modo da rendere più agevole e piacevole la manipolazione del dispositivo, e allo stesso tempo nel predisporre lo spazio per l'illuminazione si è scelto di sfruttare una forma, data dai tre esagoni, posizionata alla base del cubo, su ciascuna delle quattro facce laterali.

Per quanto riguarda l'*affordance*, ossia "l'idea che la natura delle cose possa suggerire le loro possibili utilizzazioni"[5], ad esempio l'orientamento di default del cubo è indicato dalla disposizione della griglia dello speaker, sulla faccia contenente il display, che richiama quella di alcuni smartphone e di dispositivi analoghi. La disposizione studiata per gli esagoni richiama una freccia che punta verso l'alto evidenziando l'importanza del display nel dispositivo. Inoltre la forma esagonale è stata scelta per evidenziare ulteriormente il collegamento tra questa interfaccia tangibile e il mondo delle api presentato nel cortometraggio descritto nella prima parte della tesi. Per quanto riguarda la disposizione dei tre fori esagonali, questa è stata studiata per richiamare la forma triangolare tipica dei segnali di pericolo e quindi per avvertire l'utente dell'andamento della narrazione durante i giochi previsti. Tali particolari, a livello strutturale, sono stati creati con l'intento di evidenziare la connessione tra la dimensione fisica dei cubi ed i medesimi dispositivi presenti nella dimensione virtuale rappresentata dal cortometraggio di animazione,

offrendo così all'utente la possibilità di seguire lo sviluppo della storia con maggior coinvolgimento.
Per quanto riguarda le facce laterali del cubo, è stato utilizzato un foglio rigido opaco realizzato in PVC, con uno spessore pari a 0,30 mm che offre una trasmissione della luce del 60%, diffondendo in modo ottimale la luce in presenza di una sorgente luminosa a distanza ravvicinata. Si è scelto di predisporre gli spazi per l'inserimento di tali fogli direttamente nel guscio, e non sulla struttura, in modo da ottimizzare gli spazi e rendere più solido il cubo.

Infine, dal punto di vista della modularità, si è riscontrata già a partire dai primi test la predisposizione di buona parte degli utenti, in presenza di oggetti cubici, ad avvicinare questi ultimi e combinarli, per cui si ritiene che attraverso la presenza del display e l'illuminazione delle facce laterali del cubo, l'utente non avrà difficoltà nel compiere i primi passi per sfruttare interamente le potenzialità del sistema.

Design della struttura interna

Il punto di partenza della struttura interna è stato la modellazione dei componenti elettronici utilizzati all'interno del progetto. La struttura è stata sviluppata utilizzando come sostegno la base del guscio ed è costituita da 11 parti ad incastro. La disposizione dei componenti all'interno del cubo è stata effettuata a partire dai componenti che necessitano inevitabilmente di un posizionamento ben definito nello spazio a disposizione:

- il display e lo speaker, che necessariamente avrebbero dovuto essere posizionati sopra ogni altro componente;
- la batteria che avrebbe dovuto essere appoggiata sulla base del guscio, in modo da ridurre l'uso del materiale necessario per il suo ancoraggio e distribuire in modo omogeneo il peso della struttura;
- il PowerBoost 1000 Charger e il pulsante di accensione e spegnimento del dispositivo, i quali avrebbero dovuto essere appoggiati anch'essi sulla base del guscio in quanto i fori per le rispettive funzioni di ricarica e accensione/spegnimento sono stati predisposti sulla base;
- i LED necessari per l'illuminazione delle facce avrebbero dovuto essere centrati rispetto alla forma dei tre esagoni presenti sul guscio;
- infine i due lettori NFC PN532 sono stati posizionati tenendo conto del requisito della modularità dei cubi.

I componenti rimanenti del cubo, ossia l'amplificatore audio, il Raspberry Pi Zero W e l'IMU LSM9DS1 sono stati posizionati in base alla disposizione richiesta dai

componenti appena elencati, ottimizzando lo spazio e la quantità di materiale necessaria per la stampa della struttura. Tali accorgimenti sono stati fondamentali anche per migliorare la dispersione del calore sviluppato da alcuni componenti, in particolare dalla scheda Raspberry Pi e dal PowerBoost 1000 Charger.
Per quanto riguarda i due lettori NFC, la disposizione di questi ultimi è stata studiata in modo da soddisfare il requisito della modularità che caratterizza il sistema CuBee ed è rappresentata in figura 5.5.

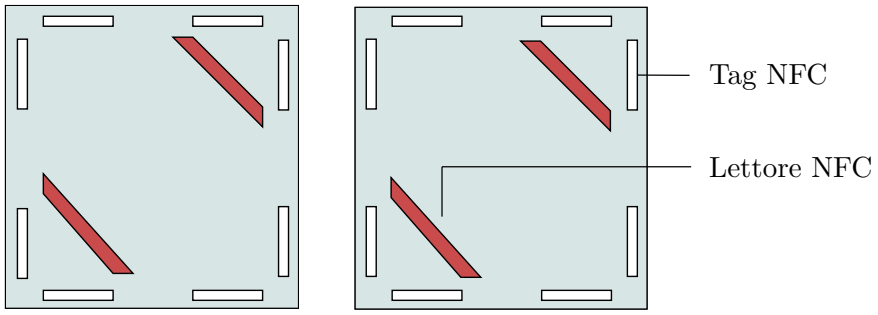


Figura 5.5: Schema della disposizione dei lettori e dei tag NFC, vista dall'alto.

I componenti appena citati sono stati opportunamente fissati alla struttura mediante viti in nylon; materiale scelto in quanto le viti potrebbero allentarsi nel tempo, rappresentando un rischio per la sicurezza degli utenti. Dal punto di vista dell'incastro delle singole parti, queste sono state progettate con un margine di tolleranza di 0,1 mm per alcune parti che necessitano di incastrarsi saldamente e 0,2 mm per quelle parti in cui è necessario avere una maggiore flessibilità. In figura 5.6 è rappresentato lo schema di montaggio della struttura.

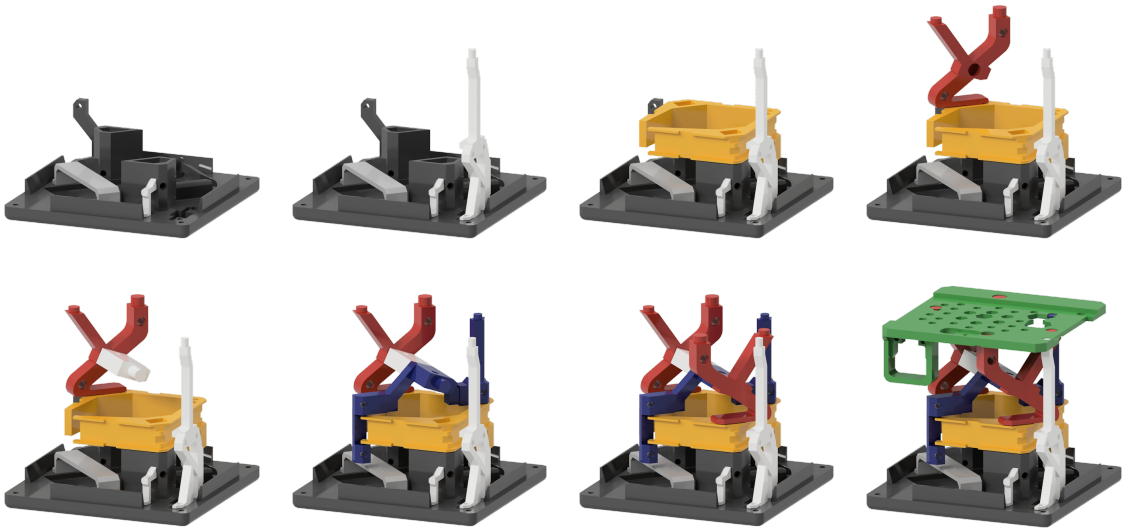


Figura 5.6: Schema di montaggio della struttura del dispositivo.

In conclusione quindi, Fusion 360 è un software estremamente articolato e ricco di funzionalità e strumenti a supporto della progettazione di prodotti e componenti di tipo industriale. Inoltre trova una crescente diffusione nella comunità dei maker grazie ad un approccio più user-friendly, fornendo strumenti utili per la stampa tridimensionale, argomento che verrà affrontato nel paragrafo successivo.

5.4 Stampa 3D

Dal punto di vista della produzione convenzionale si possono individuare tre metodi per dare vita ad oggetti solidi complessi [3]:

1. Subtractive Manufacturing, processo che consiste nel rimuovere materiale da un blocco semplice di partenza, come ad esempio la fresatura, la tornitura, le lavorazioni con le macchine a controllo numerico ecc.
2. Near Net Shape, processo che permette di ottenere la forma finale di un determinato componente, soprattutto meccanico, senza generare scarti e riducendo le fasi di finitura successive, come ad esempio lo stampaggio a iniezione.
3. Additive Manufacturing, processo che prevede la suddivisione del modello di un oggetto in strati sufficientemente sottili e la successiva realizzazione dell'oggetto fisico mediante il deposito di materiale strato dopo strato.

In quest'ultimo metodo di produzione di oggetti solidi rientra la stampa tridimensionale, in cui l'altezza dei singoli strati, o *layer*, depositati uno dopo l'altro può arrivare fino a pochi decimi di millimetro anche nel caso delle stampanti consumer. Inoltre il costo accessibile dei macchinari e dei materiali permette a chiunque di ottenere risultati impensabili fino a pochi anni fa.
Nell'ambito industriale il costo di produzione dei prototipi mediante la stampa 3D è molto più vantaggioso rispetto ad analoghe tecnologie convenzionali rientranti nell'Additive Manufacturing. Oltre ai costi ridotti, i vantaggi della stampa 3D sono dati dalla facilità di utilizzo in quanto può essere usufruita sia da piccole-medie imprese sia dal singolo individuo per uso personale, permettendo così una maggior libertà creativa; è inoltre possibile realizzare prototipi di diversa complessità in tempi relativamente ridotti, come nel caso dei dispositivi realizzati per il progetto di tesi.

Per quanto riguarda le tecnologie utilizzate nella stampa 3D, le più diffuse sono le seguenti: SLS e SLM, SLA e FDM/ FFF. Per il funzionamento di queste tecnologie, descritte in seguito, si è attinto alle informazioni riportate in [6] e [7].

Tecnologia di stampa SLS e SLM

La tecnologia di stampa SLS (Selective Laser Sintering), risale agli anni '80 ed è basata sulla sinterizzazione, ossia su quel processo termico e meccanico inteso ad ottenere materiali compatti da sostanze polverulente. Questo processo consiste nel riscaldare uno strato di polvere polimerica ad una temperatura prossima a quella di fusione, mentre il compito di portare parte di questa polvere alla temperatura di fusione è affidato ad un laser, il quale percorrendo la sezione trasversale del modello 3D porta le particelle ad unirsi dando forma ad un layer del modello. Il procedimento si ripete per ogni layer previsto, abbassando il piano di stampa e riapplicando uno strato di polvere polimerica. La granulometria del materiale di consumo ricopre un ruolo importante in quanto è direttamente collegata alla risoluzione finale dell'oggetto stampato.
La tecnologia SLM (Selective Laser Melting) è del tutto analoga ma si basa sulla fusione di polveri metalliche.

Uno dei vantaggi di questa tecnologia è la possibilità di essere utilizzata per la realizzazione di numerose parti mobili, parti interdipendenti e altri progetti complessi, in quanto non necessita di strutture di supporto. Con la sinterizzazione laser è quindi possibile stampare oggetti complessi, che invece possono risultare molto più impegnativi nel caso di una stampante FDM se non persino irrealizzabili in alcuni casi. Gli oggetti creati con questa tecnologia in genere necessitano di un trattamento successivo alla stampa in quanto risultano avere una superficie granulosa e possono essere sottoposte a rifiniture quali sabbiatura, verniciatura, rivestimenti vari ecc. Per quanto riguarda i materiali, questi possono essere di vario genere: nylon, polistirene, acciaio, titanio ecc.

Dal punto di vista dei costi le stampanti che utilizzano questa tecnologia hanno un prezzo non accessibile in ambito consumer in quanto le sole macchine partono dai 5.000 euro, superando anche quota 200.000 euro per macchine di tipo industriale. Rappresenta ad ogni modo un'alternativa economicamente conveniente rispetto allo stampaggio a iniezione.

Tecnologia di stampa SLA

La SLA (Stereolithography) o stereolitografia, inventata da Chuck Hull, co-fondatore della 3D Systems, è stata la prima tecnologia di stampa 3D ad essere brevettata, negli anni '80. Ad oggi, insieme alla FDM, rappresenta la tecnologia più diffusa in ambito consumer, grazie anche ai costi di poco superiori.
Questa tecnologia sfrutta la fotopolimerizzazione di una resina liquida che si trova all'interno di un contenitore trasparente; un laser posizionato sotto o sopra al contenitore viene opportunamente direzionato attraverso degli specchi in modo da

seguire una determinata traiettoria che ricalca la forma codificata dal software di slicing. Infine il puntamento del laser causa la solidificazione della resina nei punti interessati.

La risoluzione di stampa ottenibile attraverso questa tecnica è dell'ordine di 0,03 mm, un valore piuttosto alto, ed è anche questo uno dei suoi puoi di forza; per tale motivo è particolarmente adatta alla stampa di oggetti di dimensioni ridotte in cui il requisito è l'alta precisione.

Un'evoluzione di questa tecnologia è data dalla tecnologia DLP (Digital Light Projector) che utilizza un proiettore video digitale in sostituzione del laser ed in un singolo passo permette di proiettare la luce sufficiente per l'intero layer, rispetto alla SLA che prevede il continuo movimento del laser.

Tecnologia di stampa FDM/ FFF

La tecnica di stampa 3D basata sul deposito di filamento fuso prende il nome FDM (Filament Deposit Modelling) ed è attualmente una delle tecnologie più diffuse, soprattutto nell'ambito consumer. Si tratta di un'invenzione risalente al 1988 ed è attribuita a Scott Crump, colui che successivamente avrebbe fondato la Stratasys. Le stampanti 3D che utilizzano tale tecnica hanno avuto un vero e proprio boom a partire dal 2009 quando alcuni brevetti, tra cui quello di Crump, sono diventati di pubblico dominio. Non potendo queste utilizzare la denominazione FDM, in quanto marchio depositato, è stata introdotta la sigla FFF (Fused Filament Fabrication) per riferirsi alla medesima tecnologia.

La tecnica FDM prevede la fusione di un materiale termoplastico (definizione delle materie plastiche che acquistano modellabilità sotto l'azione del calore) in forma di filamento di spessore compreso tra 1,75 mm e 2,85 mm; questo filamento viene meccanicamente introdotto all'interno dell'estrusore ad una temperatura compresa tra i 180 °C ed i 300 °C e viene infine depositato sul letto di stampa raffreddandosi pressoché istantaneamente e dando forma, strato dopo strato, all'oggetto fisico.
Lo spessore dei singoli layer può arrivare fino a pochi decimi di millimetro, a seconda delle caratteristiche della stampante e della dimensione dell'ugello montato sull'estrusore.

Dal punto di vista della composizione le stampanti 3D basate sulla FDM possono avere uno o due estrusori, permettendo in quest'ultimo caso di poter utilizzare due filamenti diversi e quindi di poter stampare oggetti bicolore oppure oggetti particolarmente complessi avvalendosi di supporti stampati usando un filamento in materiale solubile, come ad esempio il PVA o l'HIPS. I supporti infatti, sono necessari per quelle parti dell'oggetto, definite *overhangs*, che non dispongono di uno

strato sottostante su cui il materiale, depositato dall'estrusore, possa raffreddare correttamente.

I vantaggi di questa tecnologia sono dati dal basso costo dei filamenti (a partire da 15 euro) e delle stampanti (a partire da 100 euro), quindi in definitiva dal costo di stampa; inoltre vi è un'ampia gamma di stampanti a partire da quelle più economiche DIY, ossia che devono essere assemblate dall'utente, fino ad arrivare a stampanti più robuste e affidabili, solitamente già assemblate e pronte all'uso. I filamenti disponibili per queste stampanti sono anch'essi molteplici sia in termini di colore che di materiale, con diverse composizioni contenenti ad esempio particelle di legno, carbonio ecc.

Le due stampanti utilizzate per questo progetto sono entrambe FDM e verranno approfondite in seguito.

Materiali

Le stampanti FDM possono utilizzare diverse tipologie di filamenti, con caratteristiche anche molto differenti l'uno dall'altro. I più diffusi sono i filamenti in PLA che permettono di ottenere un buon risultato a basso costo e allo stesso tempo sono caratterizzati da una discreta facilità di stampa, a differenza di materiali come l'ABS, più difficili da gestire.

Nell'ambito della stampa tridimensionale consumer i filamenti sono costituiti da materiali termoplastici e si presentano in due possibili diametri: 1,75 mm e 2,85 mm. Alcuni dei materiali che trovano utilizzo nelle bobine di filamenti per la stampa 3D FDM sono: PLA, ABS, TPU, TPE, PET, PET-G, Nylon, Policarbonato, Polipropilene. Di questi materiali i più diffusi sono il PLA (l'unico tra quelli elencati ad essere completamente biodegradabile), l'ABS, il TPU e il PETG.

Il PLA (acido lattico) è una bioplastica derivante dalla trasformazione degli zuccheri presenti nel mais. Essa ha caratteristiche simili al poliestere e al PET, ma a differenza di questi ultimi è completamente biodegradabile sotto determinate condizioni di temperatura e umidità. Oltre alla biodegradabilità, un'ulteriore caratteristica è rappresentata dalla facilità di stampa, aspetto che ha contribuito a rendere questo filamento estremamente diffuso.

L'ABS (acrilonitrile-butadiene-stirene) è invece un polimero termoplastico che ha come vantaggio la rigidità e la resistenza all'impatto. Attualmente trova utilizzo in ambiti molto diversi ed è il materiale con cui sono stati realizzati per lungo tempo i mattoncini LEGO, i quali a partire dal 2018 stanno per diventare più sostenibili [8]. È diffuso nella stampa 3D di componenti e parti che devono resistere a carichi fisici importanti.

Il TPU (poliuretano termoplastico) è un materiale flessibile, per cui la stampa con questa tipologia di filamento necessita di una maggiore attenzione e cautela, richiedendo velocità di stampa molto basse. Vi si trovano anche filamenti in TPE, ossia elastomeri termoplastici, con caratteristiche simili alla gomma e proprio tale elasticità è il motivo per cui questa tipologia di filamento è più difficile da stampare.

Il PETG rientra nella famiglia dei poliesteri e viene definito come resina termoplastica. A differenza del PET è leggermente più flessibile ed ha una maggiore resistenza agli urti. Inoltre è caratterizzato dall'alta resistenza meccanica e chimica.

Negli ultimi anni hanno preso vita molteplici tipologie di materiali che possono essere utilizzati per la stampa 3D. Infatti, oltre ai classici materiali appena citati ve ne sono altri che presentano caratteristiche particolari: per la stampa di supporti è possibile utilizzare ad esempio il PVA (alcol polivinilico) che si dissolve in acqua e l'HIPS (High Impact PolyStyrene) che è solubile in limonene. Vi sono inoltre alcuni filamenti compositi, a base di PLA:

- Legno: si tratta di filamenti in PLA ai quali vengono aggiunte delle fibre di legno in polvere o granelli. Sia dal punto di vista estetico che dal punto di vista olfattivo gli oggetti realizzati con questa tipologia di filamento, come la linea EasyWood di Formfutura, ricordano l'aspetto e l'odore del legno. La tipologia di legno, e quindi il colore, può essere diversa: pino, sughero, ebano, cocco, olivo ecc. La concentrazione di particelle miscelata al PLA è compresa tra il 10% e il 40%.
- Metallo: analogamente ai filamenti che richiamano le caratteristiche del legno, questa tipologia di materiale è costituita da PLA al quale vengono miscelate delle polveri metalliche, come alluminio, bronzo, acciaio ecc. Alcuni esempi sono i filamenti della linea MetalFil di Formfutura e Steelfill, Bronzefill e Copperfill di colorFabb.

In aggiunta a queste tipologie più diffuse vi sono altri filamenti in PLA, ABS, PETG o Nylon che vengono rinforzati con fibre di carbonio, come il filamento CarbonFil di Formfutura, i quali possono resistere a temperature elevate e all'impatto, o ai quali vengono aggiunte polveri di pietra, come i filamenti della linea StoneFil di Formfutura, ottenendo un aspetto notevole dal punto di vista estetico.
Vi sono ancora alcuni filamenti derivanti da scarti vegetali, e quindi compostabili, come i filamenti Tomato, Hemp e Weed di Kanesis, prodotti in Italia con scarti provenienti da pomodori e canapa che vengono addizionati ad una base PLA. Questi materiali sono particolarmente adatti per le prove di stampa, in quanto riducono l'impatto ambientale complessivo del processo di prototipazione.

Tali tipologie di filamenti contenenti fibre, particelle o polveri risultano più problematiche nella stampa in quanto contribuiscono ad ostruire rapidamente l'ugello dell'estrusore.

Il successo di un prodotto, nella maggior parte dei casi, dipende dalla capacità degli sviluppatori di identificare correttamente i bisogni degli utenti a cui il prodotto è destinato e realizzare tale prodotto in maniera sostenibile, soprattutto in termini economici ma sempre di più anche con un riguardo al ciclo di vita e all'impatto ambientale generato. In questo senso quindi per la realizzazione dei prototipi di questo progetto si è deciso di utilizzare un materiale biodegradabile come il PLA, scegliendo di trasmettere l'aspetto naturale del materiale attraverso l'utilizzo di un materiale composito contenente il 30% di particelle di legno.

Workflow

Per quanto riguarda il flusso di lavoro, il processo di stampa tridimensionale si articola in due fasi: (1) lo slicing del modello 3D e la generazione del file con le relative istruzioni per la stampante (GCODE) e (2) la fase di stampa effettiva. È evidente come questo processo sia estremamente snello e permetta di ridurre i tempi, i costi e la complessità coinvolta; infatti trascurando la fase di modellazione dell'oggetto, gli strumenti coinvolti sono solamente la stampante 3D e il software di slicing.

La preparazione del modello 3D da stampare viene effettuata mediante un apposito software, come Simplify3D o Ultimaker Cura, il quale effettua il cosiddetto slicing del modello a seconda dei parametri impostati dall'utente. Tale operazione, comune a tutte le tecnologie di stampa 3D, consiste nell'intersezione del modello con una serie di piani paralleli al piano di appoggio del modello, e opportunamente distanziati, per ottenere le singole sezioni che definiscono il modello.

I parametri che definiscono la qualità della stampa sono molteplici e permettono di stabilire l'altezza dei singoli layer, il pattern e la densità di riempimento, la temperatura di stampa, la retrazione, la generazione dei supporti, la velocità di stampa e di spostamento, il raffreddamento ecc. È anche per questo motivo che l'ottenimento di un oggetto qualitativamente accettabile è un'attività che richiede una notevole quantità di tempo.

Uno dei risultati più difficili da ottenere con una stampante consumer da poche centinaia di euro riguarda la precisione di stampa, che coinvolge sia gli aspetti hardware sia quelli legati al firmware della stampante. Bisogna tener conto infatti che questa tipologia di stampanti viene utilizzata per lo più in ambito domestico in cui gli errori dimensionali degli oggetti stampati possono essere accettati entro un certo limite ragionevole (di solito pochi mm) e la maggior parte delle stampanti riescono a soddisfare tali requisiti. Nell'ambito della prototipazione, utilizzata per la realizzazione di un prototipo assimilabile ad un prodotto finito, è necessaria una precisione maggiore: ad esempio per un componente di $10 \times 10 \times 20$ mm a cui è richiesta una tolleranza massima di 0,1 mm, non è accettabile avere un oggetto con

dimensioni 10,15 × 10,15 × 20,15.
Generalmente è possibile effettuare opportune modifiche ad una serie di parametri presenti nel firmware della stampante effettuando un processo di calibrazione che si articola nei seguenti passi:

1. Effettuare lo slicing di un cubo di dimensioni note, ad esempio 20 × 20 × 20.
2. Stampare il cubo di riferimento.
3. Misurare i lati del cubo stampato.
4. Sostituire il valore iniziale del parametro steps/mm del firmware della stampante, e S_{in}, per ciascun asse x, y e z, con il valore S_{fin} calcolato attraverso la seguente formula: $S_{fin} = \frac{d_T}{d_M} \cdot S_{in}$, con d_T valore teorico del lato del cubo e d_M valore misurato del lato del cubo.

Il movimento della testina di stampa lungo gli assi x, y e z è regolato da motori stepper, ed il parametro appena citato, steps/mm, si riferisce al numero di step che ognuno di questi motori stepper compie per eseguire un movimento di 1 mm lungo il proprio asse. Anche il *feeder* che fornisce il filamento all'estrusore è regolato da un motore stepper e in questo caso il valore steps/mm si riferisce al numero di step che il motore deve compiere per estrudere 1 mm di filamento.

I passi appena indicati per eseguire la calibrazione contribuiscono ad aumentare notevolmente la precisione di stampa degli oggetti, tuttavia, per quanto riguarda le prove effettuate con la stampante Anycubic i3 Mega S, la calibrazione eseguita in questo modo viene mantenuta esclusivamente su oggetti dimensionalmente simili al modello 3D utilizzato per la calibrazione, mentre per stampe di modelli con dimensioni molto diverse si sono comunque riscontrati errori legati all'accuratezza e per i quali non è stato possibile individuare una correlazione con le misure degli oggetti stampati.

Infine per quanto riguarda i formati coinvolti nel processo di stampa vi sono principalmente il formato STL (Stereolithographic file), utilizzato dal software di slicing, e il formato GCODE che è a tutti gli effetti un linguaggio macchina, utilizzato dalle stampanti 3D.
La sintassi di un file STL, formato preposto a rappresentare i dettagli geometrici di un oggetto tridimensionale, è la seguente:

```
solid OpenSCAD_Model
    facet normal -1 0 0
        outer loop
            vertex 0 0 0
            vertex 0 20 0.2
            vertex 0 20 0
```

```
        endloop
    endfacet
endsolid OpenSCAD_Model
```

Nell'esempio riportato vengono descritti i vertici di un triangolo in uno spazio di coordinate 3D. Ogni file STL è estremamente semplice e descrive nel modo appena indicato una serie di triangoli con i quali è stato approssimato il modello tridimensionale. Come si può intuire questo formato contiene quindi unicamente informazioni relative alla superficie, o *shell*, del modello. Inoltre non sono presenti informazioni relative alla texture o altre caratteristiche del modello.

Per quanto riguarda il formato GCODE, questo è molto più articolato trattandosi di un vero e proprio linguaggio e le istruzioni sono numerose. Un esempio contenente alcune istruzioni esemplificative è il seguente:

```
G28 X0 Y0 ; sposta l'estrusore nella posizione (x,y) iniziale
M140 S55  ; imposta la temperatura del piano di stampa a 55°C
G1 F240 E3 ; estrude 3 mm di filamento a velocità 4 mm/s
G1 E1 F240 ; ritrae del filamento
G1 X10 Y20 E5 ; sposta l'estrusore in (10,20) estrudendo 5 mm di
 ↪  filamento
M84  ; spegne i motori stepper
```

In conclusione, i vantaggi introdotti con la stampa 3D sono molteplici e l'applicazione di questa tecnologia ad esempio nell'ambito dell'istruzione, della ricerca e della prototipazione, offre nuove possibilità interessanti.
D'altra parte gli svantaggi, soprattutto per le stampanti consumer più economiche, riguardano il facile fallimento delle stampe per via del livellamento del piano di stampa, della temperatura sbagliata, degli errori presenti nel modello ecc., i lunghi tempi di stampa e l'alta probabilità di fallimento per stampe di lunga durata. Un ulteriore svantaggio, come accennato in precedenza, riguarda la difficoltà di ottenere stampe qualitativamente accettabili ed accurate. Inoltre è necessario generare supporti per stampare quelle sezioni del modello che non dispongono di layer di appoggio, rendendo quindi difficile stampare elementi complessi senza eventuali operazioni di post-processing.

La variabilità dei risultati ottenibili mediante la stampa 3D è strettamente correlata alla tipologia e alle caratteristiche del filamento utilizzato, e uno degli aspetti più importanti in termini di tempo impegnato in questa fase del progetto è proprio dato dalla difficoltà di ottenere un prodotto qualitativamente accettabile in modo rapido; i limiti della stampa tridimensionale infatti si riscontrano solo in fase di

utilizzo. A questo scopo sono stati necessari numerosi giorni di prove di stampa e per il cosiddetto *tuning* della stampante e delle impostazioni di stampa.

Per quanto riguarda la stampante 3D adoperata, inizialmente è stata utilizzato il modello Ultimaker 3 disponibile nei laboratori del Politecnico di Torino ma per motivi logistici e dettati dalla necessità di utilizzare materiali differenti dal classico PLA si è scelto di acquistare e utilizzare una stampante 3D differente, la Anycubic i3 Mega S, descritta nel sottoparagrafo successivo.

5.4.1 Anycubic i3 Mega S

La stampante 3D utilizzata per la produzione dei prototipi è il modello i3 Mega S, del produttore cinese Anycubic. Si tratta di una stampante consumer a tecnologia FFF e con un rapporto qualità-prezzo ottimo; infatti è stata acquistata ad un prezzo di 250 euro e attraverso una serie di miglioramenti hardware la qualità di stampa finale risulta del tutto comparabile alla ben più costosa Ultimaker 3, utilizzata per le prove di stampa iniziali.

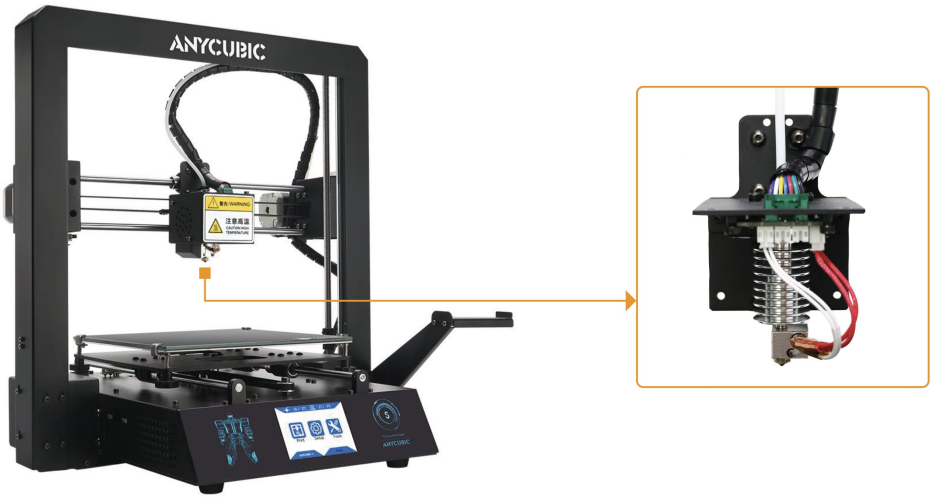


Figura 5.7: Anycubic i3 Mega S; a destra dettaglio dell'estrusore. Fonte: Anycubic.

Le caratteristiche di questa stampante, visibile in figura 5.7, sono le seguenti:

- Tecnologia di stampa: FDM/FFF.
- Volume di stampa (L×W×H): 210 × 210 × 205 mm
- Precisione di posizionamento dell'estrusore: 0,0125 mm lungo gli assi x e y e 0,002 mm lungo l'asse z.
- Altezza layer: 0,05 - 0,40 mm.

- Materiali supportati: PLA, TPU, ABS, HIPS e Nylon.
- Tipologia estrusore: Bowden.
- Velocità di stampa: 20 - 100 mm/s
- Diametro del filamento: 1,75 mm.
- Temperatura massima dell'estrusore: 275 °C.
- Temperatura massima del piano di stampa: 100 °C.
- Stampa da SD Card o porta USB.

La i3 Mega S utilizza 4 motori stepper per gestire il movimento della testina di stampa lungo gli assi x, y e z e l'estrusore. Il piano di stampa può muoversi lungo l'asse y, mentre la testina di stampa si può spostare lungo gli assi x e z. Tali motori sono gestiti da alcuni driver posizionati sulla scheda Trigorilla che si trova all'interno della stampante e rappresentano la principale fonte di rumore della stampante.

Uno degli aspetti più importanti per la riuscita di una stampa è dato dal livellamento del piano, ossia la regolazione della distanza tra il piano di stampa e l'ugello nella sua posizione iniziale. Tale distanza determina lo spessore della linea di materiale depositata dall'ugello, per cui con una distanza troppo elevata si otterrà un layer caratterizzato da una bassa adesione al piano di stampa e con gap tra le linee oltre che tra i singoli layer depositati; viceversa una distanza eccessivamente ridotta porterebbe l'ugello a contatto con il piano di stampa, danneggiando il piano e depositando un layer estremamente sottile, rischiando inoltre di ostruire l'ugello con alcuni filamenti più problematici. Il livellamento del piano di stampa è cruciale per ottenere un primo layer ottimale, fondamentale per la riuscita della stampa.

Nel caso dell'Anycubic i3 Mega S, il livellamento è manuale e a questo scopo sono presenti 4 viti agli angoli del piano di stampa, le quali permettono di abbassare e alzare il piano. La distanza consigliata tra piano e ugello è di circa 0,2 mm, tuttavia tale valore dipende dal diametro dell'ugello, in questo caso 0,4 mm, e anche dal materiale. A livello pratico per effettuare il livellamento si è utilizzato un foglio di carta semplice posizionato sul piano e sono state regolate in ordine antiorario le viti del piano in modo che l'ugello potesse scorrere liberamente con una lieve resistenza al movimento. Tuttavia questo procedimento non è stato sufficiente a garantire un buon risultato del primo layer quindi in fase di slicing è stata ridotta la velocità di stampa del primo layer, in modo da poter sistemare la distanza piano-ugello in tempo reale, interrompendo il processo di stampa e ripartendo con la nuova configurazione.
È quindi evidente che il livellamento manuale introduce di norma una serie di errori legati all'accuratezza nel processo di stampa; inoltre il livellamento del piano

deve essere eseguito frequentemente in quanto le viti e le molle che permettono di aggiustare la distanza piano-ugello tendono a cedere dopo poche stampe.

I risultati ottenuti con questa stampante sono notevoli considerato il suo costo e le possibilità di personalizzazione che permettono di migliorare ulteriormente la qualità di stampa, ridurne la rumorosità attraverso l'utilizzo di driver come i TMC2208 o nuove ventole come quelle realizzate da Noctua ecc.

Il processo di stampa in questo progetto ha richiesto un impegno notevole e ha occupato buona parte del tempo a disposizione, anche per via delle innumerevoli variabili da tenere in considerazione e di alcuni problemi avuti con questa stampante, specie nelle stampe più impegnative come quella del cubo da 88 mm che ha richiesto 23 ore di stampa e sono stati non pochi gli errori dovuti all'ostruzione dell'ugello, ad interruzioni della stampante, a problemi del filamento in legno, a errori estetici causati dal deposito di materiale in alcune parti più sensibili del cubo ecc.

Per quanto riguarda la precisione di stampa sono stati stampati quattro cubi di prova di dimensioni diverse, 20 mm, 40 mm, 60 mm e 80 mm, e si è misurata la differenza tra il lato teorico e quello del cubo stampato. I valori ottenuti sono stati utilizzati per correggere il parametro steps/mm relativo ai motori stepper degli assi x e y della stampante, come descritto nel paragrafo precedente. Nei grafici in figura 5.8 si può notare il miglioramento della precisione di stampa in seguito alla calibrazione, che ha reso più omogenei gli errori introdotti lungo gli assi x e y della stampante, sebbene tali errori tendano a mantenere un andamento non lineare.

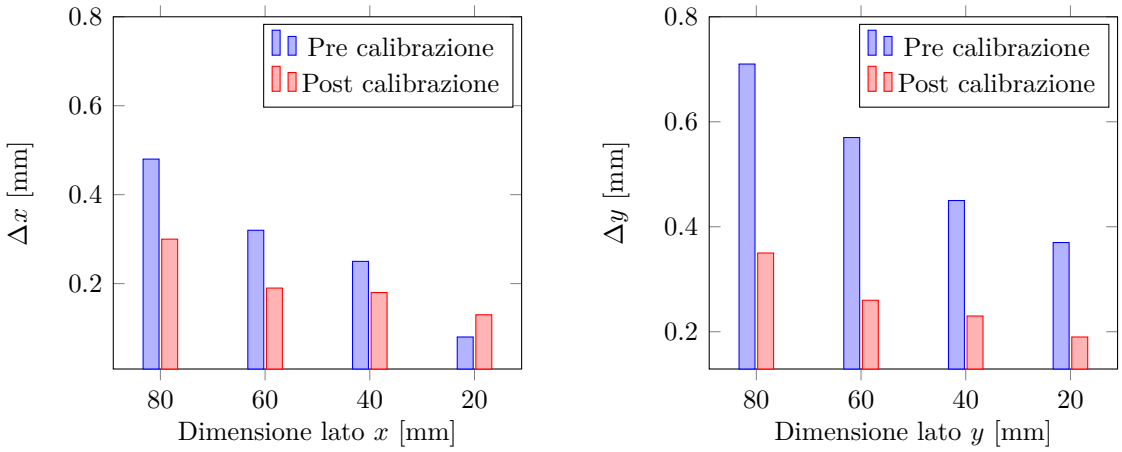


Figura 5.8: Errori di stampa relativamente agli assi x e y, prima e in seguito alla calibrazione.

Dal punto di vista del risultato finale la qualità ottenuta è molto soddisfacente, nonostante l'utilizzo di un filamento più problematico da gestire rispetto al classico PLA e gli aspetti fin qui introdotti. Alcune immagini dei risultati ottenuti sono

presenti nel paragrafo successivo, in cui verrà illustrato il prototipo conclusivo per il progetto di tesi.

5.5 L'evoluzione del dispositivo

Per quanto riguarda il sistema sviluppato occorre premettere quanto segue: innanzitutto l'obiettivo principale di questa parte della tesi è stato quello di creare un oggetto fisico, un'interfaccia tangibile, attraverso cui sperimentare le modalità di interazione adatte ad un contesto museale. In questo senso quindi non si tratta di uno sviluppo assimilabile ad un prodotto, di cui le fasi di sviluppo che si susseguono portano alla realizzazione di un oggetto completo e pronto per la commercializzazione; bensì rappresenta una fase di studio e di valutazione nella quale si intende porre maggior interesse al percorso creativo e progettuale per arrivare ad un risultato, in qualunque forma esso sia.
Pertanto l'obiettivo è stato quello di creare un prototipo funzionale rispondente a determinati parametri fisici ed estetici, e verificare il riscontro degli utenti, nello specifico dei visitatori di un museo.

Il dispositivo conclusivo per questo progetto di tesi rappresenta l'evoluzione di più di 20 prototipi realizzati nell'arco di pochi mesi. Il motivo di tale numero è dato dalle necessità di modifiche in seguito al cambiamento di alcuni componenti elettronici e quindi delle diverse soluzioni adottate nel tempo, anche in seguito ad alcuni test effettuati con gli utenti.

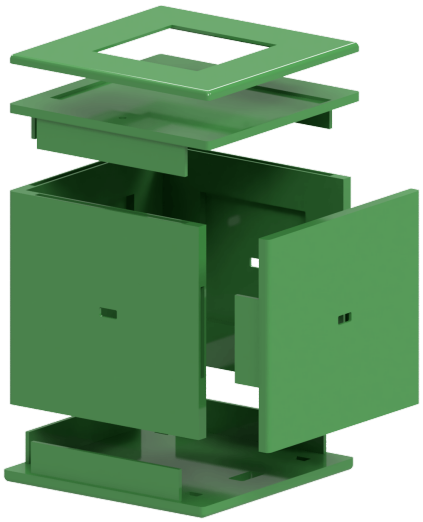


Figura 5.9: Render della prima versione del cubo.

Il prototipo iniziale del dispositivo, risalente alle prime settimane di vita del progetto, è illustrato in figura 5.9. Si trattava in questo caso di un cubo di dimensioni 80×80×80 mm e suddiviso in cinque parti da comporre mediante un meccanismo ad incastro. Tale esigenza di scomposizione del cubo è derivata da alcune problematiche sorte in questa fase, di cui la principale riguarda i tempi di stampa superiori alle 23 ore e l'alta probabilità di interruzione della stampa in corso per via di errori della stampante 3D o per cause esterne.
Per quanto riguarda la funzione di modularità si era inizialmente valutato l'utilizzo degli infrarossi, pertanto il prototipo disponeva di un foro collocato in modo da ospitare il trasmettitore e il ricevitore IR.

Questo primo prototipo è stato realizzato con Rhinoceros e siccome è stato stampato presso i laboratori del Politecnico di Torino con la stampante Ultimaker 3, non è stato possibile utilizzare un materiale diverso dal PLA. Anche per questo vincolo sul materiale utilizzabile si è in seguito deciso di effettuare un modesto investimento ed acquistare la stampante Anycubic i3 Mega S descritta precedentemente.

Dal punto di vista dei requisiti descritti nel paragrafo 5.1, si può notare come vi sia un problema legato all'illuminazione delle facce del cubo; infatti inizialmente non erano previsti dei fori a tale scopo, bensì si era realizzato un quadrato incavato, con lato di 45 mm e spessore di 1,20 mm rispetto allo spessore dei lati del guscio pari a 4,50 mm.
Per quanto riguarda il display del dispositivo, inizialmente si era previsto l'utilizzo di un altro modello che tuttavia è stato in seguito scartato per motivi legati alla compatibilità con la scheda Raspberry Pi.

Altra caratteristica di questi primi modelli riguarda la componentistica interna ancora non del tutto definita e testata per cui essa è anche la causa delle modifiche successive alla struttura esterna del cubo, mentre dal punto di vista dei materiali anche questo aspetto è stato in seguito modificato, optando per un materiale e un colore diverso dal PLA bianco utilizzato inizialmente.

Nei prototipi successivi al primo, i cambiamenti hanno riguardato la modifica di alcuni dei componenti utilizzati e conseguentemente la loro disposizione all'interno del dispositivo. Dal punto di vista dell'aspetto ergonomico ed estetico sono state modificate alcune caratteristiche del cubo in base ai feedback avuti dagli utenti che hanno testato le prime versioni del dispositivo. Nella figura 5.10 sono riportati alcuni dettagli di tali versioni.

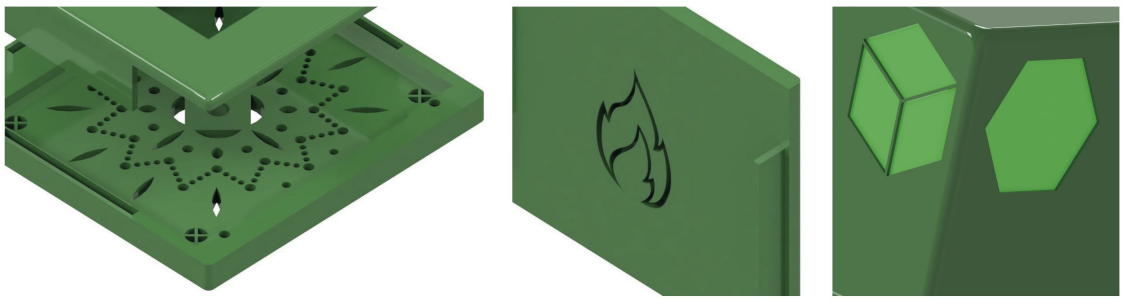


Figura 5.10: A partire da sinistra dettagli di alcuni prototipi iniziali: alloggiamento del display, faccia laterale e diverse forme illuminate sulle facce laterali del cubo.

Caratteristiche del prototipo finale

Il guscio del prototipo finale è mostrato nella figura 5.11. Innanzitutto le dimensioni del dispositivo sono aumentate del 10% rispetto alla prima versione, incremento

reso inevitabilmente necessario dal cambiamento del display utilizzato, il quale ha richiesto una dimensione maggiore del dispositivo per poter essere incorporato. Le dimensioni attuali del cubo rimangono ad ogni modo nel limite della manipolabilità, consentendo un utilizzo agevole dei dispositivi.

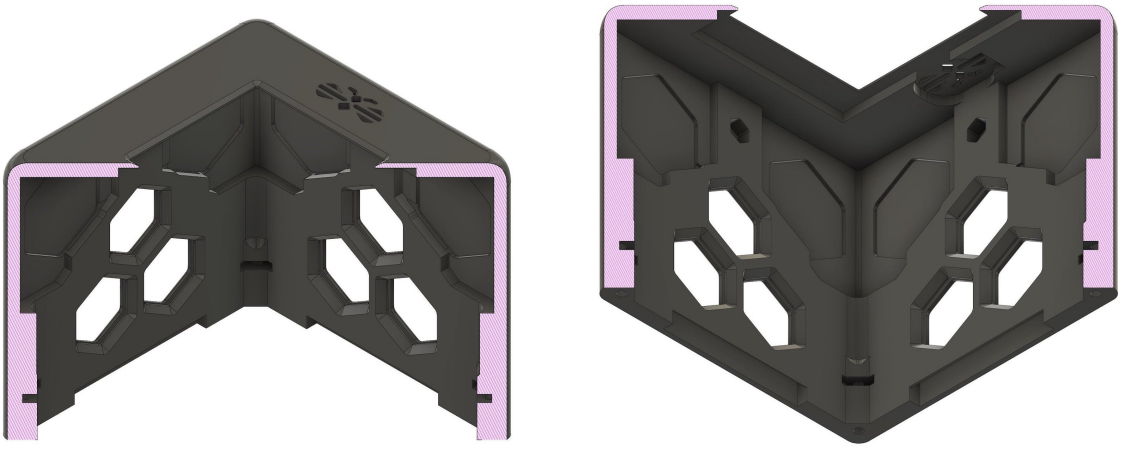


Figura 5.11: Sezioni del guscio del prototipo finale in cui si possono notare gli alloggiamenti per i tag NFC, per i magneti, per lo speaker e per le viti di chiusura del cubo.

In questo ultimo prototipo sono stati predisposti gli alloggiamenti per l'inserimento di piccoli magneti al neodimio, i quali sono stati inseriti per facilitare il collegamento dei cubi e mantenerli in posizione salda. Allo stesso tempo il meccanismo di aggancio associato, consistente nella forza di attrazione dei magneti e in parte nel suono associato a tale aggancio, contribuisce ad evidenziare all'utente le dinamiche di gioco.

Dal punto di vista multisensoriale è ampiamente condivisa in letteratura la teoria secondo cui nel contatto percettivo con un determinato manufatto il tatto assume spesso caratteristiche predominanti sugli altri sensi, in quanto "la modalità che elicita la maggior sensazione di piacevolezza appare essere attribuibile proprio a questo senso" [9].
L'importanza della sensazione tattile nel dispositivo è stata tenuta in considerazione attraverso quelli che Eleonora Buiatti in [9] definisce "proprietà macrogeometriche", ossia quegli stimoli che riguardano la forma, la struttura e le dimensioni dell'oggetto, e "aspetti microgeometrici", relativi alla superficie dell'oggetto, come la consistenza, la ruvidità, la rigidità ecc. Infatti la differenza più evidente del prototipo finale rispetto alle prime versioni del cubo è data proprio da alcune peculiarità che influiscono in modo positivo sulla percezione del dispositivo da parte degli utenti.
Si è quindi scelto di realizzare il guscio del dispositivo in un materiale composito, nello specifico un filamento contenente il 30% di particelle di legno di pino riciclato, aggiunto ad una base di PLA. Questa tipologia di materiale è stata particolarmente

complessa da gestire, anche per via del lungo tempo di stampa richiesto dal guscio, causa di innumerevoli intasamenti dell'ugello della stampante che ha richiesto l'adozione di opportune soluzioni. Tuttavia, il modello ottenuto, visibile in figura 5.14, risulta estremamente piacevole al tatto, richiamando i pregi del legno tra cui la ruvidità e l'irregolarità caratterizzante, oltre alla naturalezza conferita al dispositivo. Un'ulteriore motivazione della scelta di questo tipo di materiale è stata la volontà di replicare il più possibile gli elementi presenti nel cortometraggio di animazione, in quanto il cubo vi si ritrova anche a livello narrativo ed ha origine proprio dalla ramificazione di un albero.

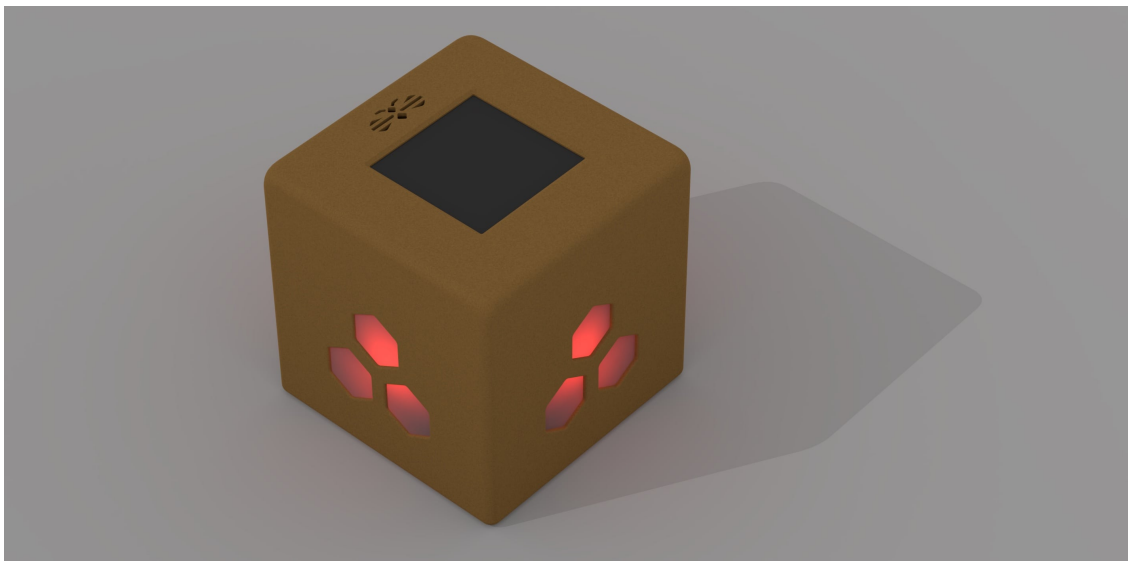


Figura 5.12: Render del prototipo finale del cubo

Tale materiale simil legno conferisce al dispositivo un valore aggiunto rispetto ai primi prototipi, in cui la consistenza *plastica* del guscio non si integrava adeguatamente nel contesto dell'installazione progettata. Inoltre alcuni utenti hanno trovato che il cubo realizzato con questo tipo di materiale composito risultava più morbido al tatto ed è stato apprezzato maggiormente rispetto allo stesso cubo stampato con un filamento in PLA. Allo stesso tempo anche la sonorità associata allo spostamento, all'appoggio e alla manipolazione dei cubi sul tavolo, è stata ritenuta più piacevole e "dolce" rispetto allo stesso cubo in PLA.

La struttura per l'alloggiamento dei componenti elettronici nel prototipo finale è riportata nella figura 5.13, mentre i cinque cubi stampati e assemblati sono mostrati nella figura 5.14.

Figura 5.13: Struttura per l'alloggiamento dei componenti elettronici nel prototipo finale.

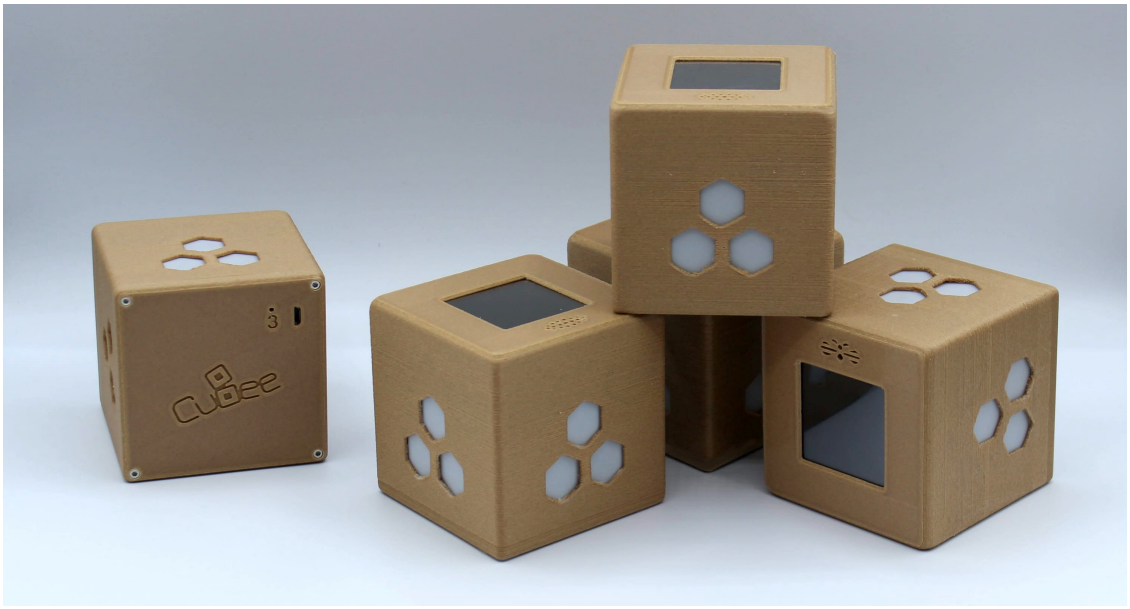


Figura 5.14: I cinque cubi del sistema CuBee stampati e assemblati.

Capitolo 6

Progettazione hardware

— Stelian Spulber

Fin dagli albori dell'informatica, si è effettuata una distinzione tra ciò che viene definito hardware e il software. L'hardware di un computer altro non è che un insieme di circuiti elettrici, componenti e vari dispositivi che hanno il compito di far funzionare l'elaboratore. In genere con questo termine si indica non solo il contenitore della macchina, il cosiddetto *chassis*, ma anche tutte le periferiche, ovvero tutti quei dispositivi che vengono collegati all'unità centrale, in modalità wired o wireless.

In questo capitolo verranno affrontati gli aspetti strettamente correlati all'hardware del sistema CuBee. Saranno quindi indicati e descritti i componenti utilizzati, tenendo in considerazione i requisiti che i dispositivi devono soddisfare, nonché i limiti tecnici ed economici riscontrati nelle fasi di progettazione e sviluppo.

6.1 Elaborazione: dai microcontrollori ai microcomputer

Generalmente un dispositivo elettronico svolge tre funzioni di base: (1) riceve una serie di istruzioni dall'utente, (2) esegue le istruzioni ricevute, elaborando opportunamente le informazioni in ingresso e (3) fornisce all'utente l'output e/o i risultati dell'elaborazione.

L'elaborazione, con riferimento a dati numerici o informazioni multimediali di vario tipo, è quindi la fase intermedia di un processo nel corso della quale un elaboratore elettronico trasforma i dati in ingresso secondo appropriate regole.

Tale concetto è rappresentato in figura 6.1, dove il segnale x rappresenta il segnale da elaborare, il blocco $E(x)$ rappresenta invece il blocco di elaborazione dei segnali in ingresso e il segnale y costituisce il risultato di tale elaborazione.

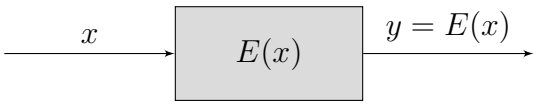


Figura 6.1: Generico processo di elaborazione di un segnale x.

All'interno dei dispositivi facenti parte di CuBee sono presenti vari componenti che generano un flusso costante di dati relativi al cubo, quali ad esempio l'accelerazione, la rotazione e la presenza o meno di altri cubi connessi. Inoltre ogni cubo riceve una serie di dati dagli altri cubi. La mole di dati disponibile ad ogni singolo cubo aumenta nell'arco di pochi secondi, pertanto è necessario elaborare tali dati effettuando un opportuno filtraggio, scartando i dati ridondanti o ininfluenti in un certo arco temporale e rendendoli significativi nella logica del gioco in esecuzione.

La gestione delle funzionalità del sistema, esplicitate attraverso i componenti elettronici presenti all'interno dei dispositivi, nonché l'elaborazione dei dati generati all'interno del sistema, devono essere effettuate da un unico dispositivo e tale scelta non può che ricadere su un computer. Si rende quindi necessario dotare ogni singolo cubo di un'unità centrale che svolga in modo efficiente i compiti cui esso deve assolvere gestendo i flussi di dati interni o esterni che scorrono al suo interno.

L'elaboratore elettronico, più comunemente detto computer, è uno strumento ormai onnipresente nella nostra vita quotidiana, dall'ambito domestico all'ambito lavorativo. Con questo termine si fa riferimento non solo a quel dispositivo portatile o meno che che si trova sulle nostre scrivanie, ma anche ad un mainframe, ad un microcontrollore, come Arduino Nano 33, oppure ad un computer dalle dimensioni di pochi millimetri, dotato di tutti i componenti necessari al suo funzionamento su un'unica scheda, come ad esempio Raspberry Pi Zero W. Le differenze certamente non sono limitate alle dimensioni, ma sono diverse le caratteristiche e le funzionalità così come gli impieghi d'uso.
Al fine di questa tesi interessano in particolar modo i microcontrollori e soprattutto l'ecosistema Raspberry Pi.

Microcontrollori

Un microcontrollore viene spesso definito come un computer su un singolo chip. Infatti i microcontrollori dispongono di un processore simile alla CPU di un computer tradizionale, il quale effettua le operazioni comunemente svolte da un microprocessore *general purpose*. Tuttavia esso può essere più correttamente definito come un

dispositivo *specific purpose*, con capacità di calcolo limitata e pensato per assolvere a specifici compiti, all'interno di contesti ben definiti (applicazioni embedded).

Oltre alla memoria RAM e FLASH, un microcontrollore incorpora nel suo chip anche una serie di elementi ausiliari per l'interfacciamento con altri dispositivi ad esempio tramite I^2C, SPI o 1-Wire, grazie alle linee di ingresso/uscita di cui dispone. Inoltre la maggior parte dei piedini del microcontrollore, generalmente, può essere programmata dall'utente.

La capacità di calcolo di questi dispositivi è molto limitata, anche se negli ultimi anni le loro prestazioni sono decisamente migliorate. Per quanto riguarda la programmazione, un microcontrollore può essere programmato in molteplici linguaggi; ad esempio le schede Arduino utilizzano il linguaggio Wiring, derivato dal C e dal C++.

Il primo microcontrollore messo in commercio, nel lontano 1971, è stato il 4004, rappresentato in figura 6.2 in una pubblicità di Intel dello stesso anno, creato da Federico Faggin per la Intel [1] e inizialmente sviluppato per la calcolatrice Busicom 141-PF. Si trattava di un microprocessore a 4 bit con massima frequenza di clock di 740 kHz, una memoria indirizzabile di 640 bytes e 16 registri general purpose; il tutto grazie a 2.300 transistor. Poiché il 4004 includeva anche le funzionalità di controllo della memoria e degli I/O, si può parlare a tutti gli effetti di microcontrollore e non solo di CPU. Il suo successore, l'8008 a 8 bit, avrebbe messo le basi per i primi personal computer.

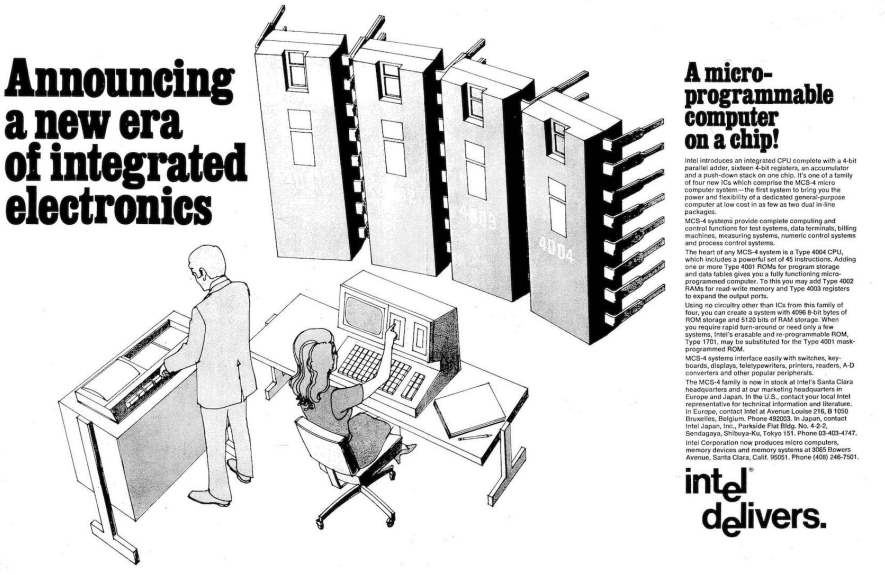


Figura 6.2: Pubblicità del microcontrollore 4004, del 1971. Fonte: Intel.

Per quanto riguarda il mercato hobbista, il microcontrollore che ne sarebbe diventato l'elemento portante, per via del basso costo, della disponibilità e della proliferazione di strumenti di programmazione, è il PIC della Microchip Technology, commercializzato a partire dal 1975 e ancora oggi disponibile.

Nei tempi recenti l'esempio di microcontrollore per antonomasia è Arduino. L'obiettivo del progetto Arduino è quello di permettere una prototipazione rapida ed economica per sistemi embedded. Ha contribuito quindi a trasformare problemi complessi legati all'hardware in problemi molto più semplici gestibili tramite software. La sua storia inizia nel 2001, con la nascita dell'Interaction Design Institute a Ivrea, nato dalla collaborazione tra Olivetti e Telecom Italia. La prima scheda Arduino prende forma nel 2005: si trattava di una scheda molto economica, dalle dimensioni di pochi centimetri e con la possibilità di usare sensori, attuatori e altri componenti elettronici programmabili in modo semplice. Se Linux aveva già posto le basi per la condivisione libera del software, per quanto riguarda l'hardware non vi erano ancora esempi rilevanti. Con l'arrivo di Arduino viene dimostrato che anche l'hardware può diventare open source, riuscendo a mantenere un equilibrio tra gli ideali di una comunità e gli aspetti puramente economici. La filosofia dell'open source è probabilmente una delle principali motivazioni del successo di Arduino.

> "La filosofia dell'apertura totale, dell'assenza di brevetti se non sul nome porta sempre più persone a modificare la scheda, migliorarla e a condividere le proprie idee." [2]

Con 24 schede prodotte dal 2005, oltre a decine di *shields*, ossia elementi che possono essere aggiunti su una scheda per aggiungere ulteriori funzionalità, Arduino è diventato il punto di riferimento dell'hardware in molteplici ambiti, dall'ambito educational ai cosiddetti maker.

Microcomputer

Differenti caratteristiche e usi trova invece il Raspberry Pi, che è un computer a tutti gli effetti e talvolta viene anche indicato con il termine microcomputer, per via delle sue dimensioni. Si tratta di un computer costituito da un'unica scheda elettronica, definito anche con l'acronimo SBC, ovvero *Single Board Computer*, sulla quale sono presenti tutti i componenti necessari per il suo funzionamento. Con più di sei anni di ricerca e progettazione alle spalle, Raspberry Pi è stato reso disponibile sul mercato nei primi mesi del 2012 [3]: Model B, la prima versione, aveva un costo di 22 sterline.

L'obiettivo del progetto ideato dalla Raspberry Pi Foundation è stato quello di promuovere l'insegnamento dell'informatica e della programmazione nelle scuole e nei paesi in via di sviluppo [4]. Tra il 2012 e l'inizio del 2018 sono stati venduti

oltre 19 milioni di Raspberry Pi [5], contribuendo a quella che è la mission della fondazione, riportata sul proprio sito, ossia:

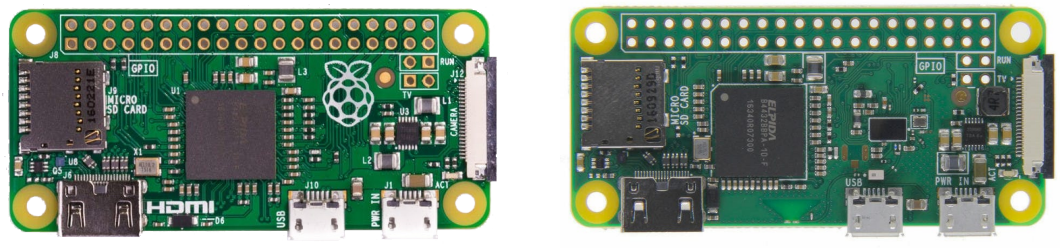
> "...to put the power of computing and digital making into the hands of people all over the world. We do this so that more people are able to harness the power of computing and digital technologies for work, to solve problems that matter to them, and to express themselves creatively."

Nel corso degli anni sono state lanciate molte versioni del Raspberry Pi, con velocità del processore da 700 Mhz a 1,5 GHz per l'ultimo arrivato Pi 4, memoria RAM da 256 MB fino a 4 GB, porte USB per la connessione di periferiche e uscita video HDMI. L'ultimo arrivato, nonché attuale modello di punta della famiglia Raspberry Pi, è il Pi 4. Questo microcomputer lanciato nel 2019 dispone di un SoC Broadcom BCM2711 con un processore quad-core a 1,5 GHz, a 64 bit, ARM Cortex-A72 e memoria RAM configurabile fino a un massimo di 4 GB. Inoltre può gestire fino a 2 display 4K tramite le due porte Micro HDMI. Il prezzo attuale della scheda va dai 40 euro per la versione con 1 GB di RAM ai 60 euro per la versione con 4 GB.

La differenza tra Arduino e Raspberry è evidente. I requisiti dei dispositivi di gioco progettati in questa tesi, in particolar modo quelli relativi alla generazione della grafica dovuta alla presenza di un display, ha fatto propendere per l'utilizzo di un microcomputer e la scelta effettuata verrà approfondita nel paragrafo successivo.

6.1.1 Raspberry Pi Zero W

Nel 2015 è stato presentato e reso disponibile sul mercato il più piccolo microcomputer Raspberry: il Pi Zero [6], figura 6.3a. Questa scheda elettronica dalle dimensioni di appena $65 \times 30 \times 5$ mm e dal costo piuttosto simbolico di 5 dollari, è stata accolta con grande entusiasmo dagli appassionati di elettronica e informatica. Il raggiungimento dell'obiettivo della Raspberry Pi Foundation, ossia rendere la programmazione accessibile a tutti, non è mai stato così vicino.



(a) **(b)**

Figura 6.3: (a) Pi Zero e (b) Pi Zero W. Fonte: Raspberry Pi Foundation.

Due anni più tardi, nel 2017, è stato lanciato il Pi Zero W, figura 6.3b. Entrambi

questi dispositivi hanno dimensioni dimezzate rispetto alle versioni più potenti, il Model A e il Model B. Su entrambe le schede si trova lo stesso SoC Broadcom BCM2835 del primo Raspberry Pi, ma con una frequenza di clock del processore di 1 GHz, e 512 MB di RAM (equivalente alla memoria dell'iPhone 4S).
All'interno del BCM2835 si trova inoltre la GPU VideoCore IV. Questa supporta le specifiche OpenGL ES 2.0 e OpenVG 1.1 e può decodificare e riprodurre video ad alta definizione a 60 fps, utilizzando il codec H.264.

Il Pi Zero W ha le stesse caratteristiche del Pi Zero ma aggiunge nuove funzionalità per quanto riguarda la connettività: Wi-Fi, Bluetooth 4.1 e Bluetooth Low Energy. Queste nuove caratteristiche hanno fatto lievitare il costo della scheda a circa 10 dollari.

Le caratteristiche del Raspberry Pi Zero W sono le seguenti:

- CPU single-core a 1 GHz;
- GPU VideoCore IV a 250 MHz;
- LPDDR2 SDRAM di 512 MB , condivisa con la GPU;
- Cache L1 di 16 KB e Cache L2 di 128 KB;
- Connettore Mini HDMI;
- Porta MicroUSB OTG;
- Slot microSD;
- Porta di alimentazione MicroUSB;
- 40 pin multifunzione: 28 GPIO, 11 SPI, 4 I^2C, 2 UART.

Il cuore del Raspberry Pi Zero W, come accennato sopra, è il SoC BCM2835 di Broadcom [7]. Su questo chip che incorpora CPU, RAM e GPU, sono presenti anche alcune periferiche tra cui PCM/ I^2S, SPI, I^2C, PWM e GPIO e la Floating Point Unit VFPv2 della ARM. All'interno del SoC si trova lo stesso processore presente nell'iPhone 3GS, l'ARM1176JZF-S della famiglia dei processori ARM11 che impiegano il set di istruzioni ARMv6.

Per quanto riguarda l'elaborazione delle istruzioni da parte di un processore, questa si articola in cinque fasi:

1. Instruction Fetch: recupero/lettura dell'istruzione dalla memoria.
2. Instruction Decode: decodifica dell'istruzione e lettura operandi dai registri.
3. Execute: esecuzione dell'istruzione.
4. Memory: attivazione della memoria, a seconda dell'istruzione.

5. Write Back: scrittura del risultato nel registro opportuno.

Al contrario della pipeline di una classica CPU, che prevede cinque stadi, la pipeline del processore ARM1176JZF-S, figura 6.4, è costituita da otto stadi:

- due stadi di lettura dell'istruzione;
- uno stadio di decodifica dell'istruzione;
- uno stadio di rilascio dell'istruzione;
- tre stadi di esecuzione dell'istruzione e uno stadio di scrittura del risultato.

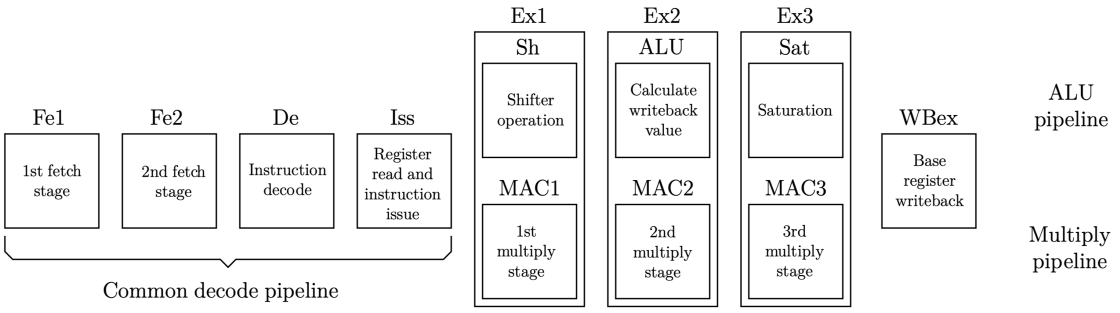


Figura 6.4: Operazioni svolte in ciascuno degli stadi della pipeline ALU. Fonte: ARM.

L'aumento degli stadi consente di aumentare la frequenza di clock, a discapito dell'introduzione di criticità per quanto riguarda i salti condizionati. Sovrapponendo le varie fasi operative, l'ARM1176JZF-S massimizza la frequenza di clock raggiungibile per l'esecuzione di ciascuna istruzione e il throughput si avvicina a 1 istruzione per ciascun ciclo di clock. Poiché questo processore ARM a 32 bit può girare ad una frequenza massima pari a 1 GHz, il computer è capace di eseguire teoricamente fino a 1 miliardo di istruzioni al secondo.

Il consumo di corrente della scheda, la quale necessita di una tensione di alimentazione di 5 V, è di circa 150 mA senza alcuna periferica connessa al dispositivo e senza programmi in esecuzione. Attività quali la riproduzione di video, l'esecuzione di programmi, la connessione Wi-Fi e Bluetooth, nonché la connessione di componenti elettronici o di periferiche possono incrementarne in modo significativo il consumo.

Il Pi Zero W, al pari di altri microcomputer e microcontrollori, supporta le modalità di comunicazione seriale SPI e I^2C, approfondite nel prossimo paragrafo. Sulla scheda sono presenti tre controllori SPI dei quali solo due, SPI0 e SPI1, sono accessibili all'utente. Di questi ultimi, SPI0 può gestire fino a due componenti o slave e SPI1 ne può gestire fino a tre. Per quanto riguarda I^2C, il numero teorico di componenti gestibili dalla scheda è di 127, fermo restando il limite fisico dovuto al numero di pin presenti sulla scheda cui collegare tali componenti.

Una delle caratteristiche più importanti della scheda Raspberry è la presenza dei pin GPIO (General-Purpose Input/Output). Tutti i pin possono essere designati, a livello software, come input o output e possono essere utilizzati per una vasta gamma di scopi.
Sulla scheda sono presenti due pin di alimentazione a 5 V e due pin a 3,3 V oltre a sette pin di Ground (0 V), che non possono essere destinati ad altro uso. I rimanenti 28 pin sono invece di tipo general-purpose.

Oltre ad utilizzare i pin GPIO come dispositivi di input e output, questi possono essere utilizzati per una varietà di funzioni alternative, alcune disponibili su tutti i pin, mentre altre solo su determinati pin. Ad esempio si hanno 11 pin relativi all'interfaccia SPI, 4 pin relativi all'interfaccia I^2C e 4 pin per il collegamento di dispositivi che utilizzano la modulazione PCM (Pulse-Code Modulation). Diversamente dagli esempi appena citati, la funzione PWM (Pulse-Width Modulation) è disponibile su tutti i pin a livello software, mentre per il suo utilizzo a livello hardware si devono usare 4 pin specifici: GPIO12, GPIO13, GPIO18 e GPIO19.

I componenti elettronici presenti all'interno del dispositivo di gioco sviluppato, elencati nei paragrafi successivi, sono saldati direttamente al Raspberry Pi laddove non vi siano più piedini che devono essere collegati allo stesso pin della scheda. In quest'ultimo caso i collegamenti vengono effettuati utilizzando una piccola scheda di prototipazione.
Queste scelte sono state effettuate per motivazioni puramente economiche, avendo un costo importante la realizzazione di un PCB personalizzato. Inoltre, trattandosi di un prototipo, si è preferito concentrare le poche risorse disponibili su aspetti più rilevanti, nell'ottica di creare una prima versione del dispositivo che rispondesse in modo efficace agli obiettivi posti in fase di progettazione.

Raspberry Pi, a differenza di un microcontrollore, necessita di un sistema operativo che ne gestisca le funzionalità e le periferiche. Raspbian è il sistema operativo ufficiale per le schede Raspberry, nonché il più diffuso, ed è basato sulla distribuzione Linux Debian. Viene rilasciato a partire dal 2015 dalla Raspberry Pi Foundation, mentre l'ultima versione, Buster, risale all'estate 2019.
Oltre all'interfaccia a riga di comando (CLI), Raspbian utilizza anche l'interfaccia grafica PIXEL, acronimo di Pi Improved X-Window Environment, Lightweight. Questa GUI si basa su un ambiente desktop LXDE (basato a sua volta sul gestore grafico X11) e il gestore di finestre impilabili Openbox.

Un ulteriore vantaggio nell'utilizzare Raspberry Pi riguarda la discreta documentazione a disposizione, i forum e la community che si è creata negli anni. Certamente vi sono computer single-board con caratteristiche e capacità computazionali migliori, ma il valore aggiunto di questo dispositivo è dato anche dagli aspetti di cui sopra.

6.2 Comunicazione seriale

Molteplici componenti elettronici possono essere collegati ad un microcontrollore o ad un computer come il Raspberry Pi tramite i pin di cui questi dispositivi dispongono. Alcuni componenti funzionano senza necessità di alcuna configurazione e regolano le proprie funzioni su una logica binaria del tipo ON/ OFF oppure utilizzando una modulazione (PWM). Ad esempio è possibile leggere un valore compreso tra 0 e N da un sensore di Hall che genera un'opportuna tensione in uscita, la quale viene a sua volta letta dal dispositivo cui è collegato e convertita in valore numerico. Oppure si può gestire l'accensione di un LED tramite semplici istruzioni che generano una tensione sufficiente ad alimentare il componente.

Componenti più elaborati di un componente monofunzione devono generare più tipologie di dati e/o dati diversi relativi allo stesso istante di tempo. È quindi necessario gestire il flusso di dati generato da, e per, un componente in modo che i dispositivi coinvolti in un collegamento fisico possano comunicare e quindi scambiarsi informazioni.
La gestione di tali flussi di dati è regolata dai protocolli di comunicazione, che fanno sì che dei dispositivi elettronici possano effettivamente comunicare l'uno con l'altro seguendo delle regole condivise e definite dal protocollo di comunicazione implementato.

I dati scambiati tra più dispositivi elettronici possono essere trasmessi in forma:

- seriale: i dati vengono inviati attraverso il collegamento fisico un bit alla volta, in sequenza (figura 6.5b).
- parallela: n bit di dati vengono inviati tutti nello stesso istante attraverso n linee di collegamento (figura 6.5a).

All'atto pratico la comunicazione in parallelo permette di ottenere prestazioni migliori in quanto è possibile ad esempio, con 8 linee di collegamento, inviare 8 bit in un solo ciclo di clock, mentre con la comunicazione seriale gli stessi dati impiegano 8 cicli di clock, utilizzando però una sola linea di collegamento.

Una linea di comunicazione si dice full duplex se è in grado di ricevere e trasmettere contemporaneamente, quindi sono presenti due canali: uno in ingresso e l'altro in uscita. Si dice invece half duplex se tale linea non può trasmettere e ricevere allo stesso tempo, avendo quindi un solo canale su cui i dati possono viaggiare, in un senso o nell'altro.

In elettronica i protocolli di comunicazione seriale più comuni sono i seguenti:

- Serial Peripheral Interface (SPI)
- Inter-Integrated Circuit (I^2C)

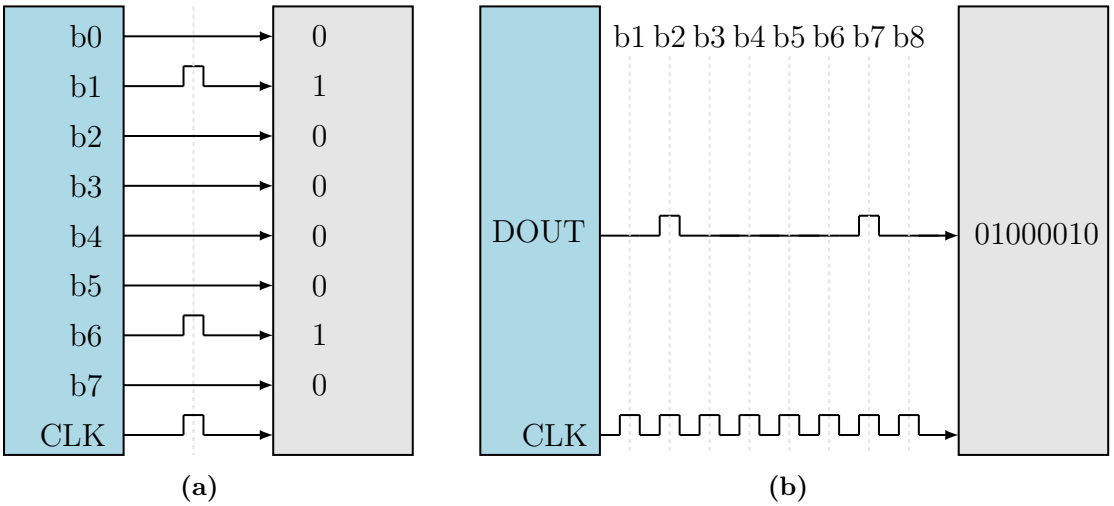


Figura 6.5: Trasmissione della lettera B = 01000010_2: (a) in parallelo e (b) in seriale.

- Universal Asynchronous Receiver/ Transmitter (UART)

Si tratta di protocolli decisamente più lenti di USB, Ethernet, Bluetooth e Wi-Fi, ma la loro implementazione è più semplice e utilizzano meno risorse hardware e di sistema.

Qualsiasi protocollo di comunicazione, tra cui SPI e I^2C, in cui i dispositivi condividono un segnale di clock è noto come sincrono. Altri protocolli sono invece asincroni come ad esempio UART, in cui i due dispositivi coinvolti nella comunicazione vengono impostati entrambi su una velocità di trasmissione preconfigurata (*baud rate*) che determina i tempi della trasmissione dei dati.

Nel protocollo SPI i dati possono essere trasferiti senza alcuna interruzione: i bit vengono inviati o ricevuti in un flusso continuo. Invece, nei protocolli I^2C e UART i dati vengono inviati in pacchetti, o frame, ognuno contenente un numero limitato di bit.

Nei successivi sottoparagrafi verranno brevemente descritti i due protocolli impiegati per la comunicazione di alcuni dei componenti presenti nei cubi: SPI e I^2C.

6.2.1 SPI

SPI è uno dei protocolli di comunicazione più diffusi che può essere utilizzato per interfacciare un componente elettronico con un microcontrollore. Si tratta di un protocollo di comunicazione seriale sincrono che permette di scambiare dati tra uno e più dispositivi a velocità elevate ($10 - 20$ Mbit/s). L'utilizzo del termine

"protocollo" viene qui usato impropriamente, in quanto al momento non esiste alcuno standard che ne definisca regole e caratteristiche. Tuttavia esistono delle regole pseudo-standard che sono state definite negli anni e che vengono seguite nonostante la loro mancata formalità.

Alla base di questo protocollo vi è il concetto di master-slave, ovvero i dispositivi collegati sono definiti master o slave e agiscono quindi diversamente. Il master è il dispositivo di controllo, ad esempio la scheda Raspberry Pi, mentre gli slave sono quei dispositivi, quali un sensore o un display, che ricevono le istruzioni dal master. Il protocollo SPI prevede un solo master, mentre non vi è un limite teorico al numero di slave possibili, sebbene questo sia limitato a pochi dispositivi nel caso di Raspberry Pi Zero W e nella maggior parte dei dispositivi.

Per gestire la comunicazione tra master e slave vengono utilizzate quattro linee, come rappresentato nello schema in figura 6.6:

1. MOSI (Master Output/ Slave Input): linea attraverso cui il master invia dati ad uno o più slave.
2. MISO (Master Input/ Slave Output): linea attraverso cui uno o più slave inviano dati al master.
3. SCLK (Serial CLocK): fornisce il segnale di temporizzazione.
4. SS, CS o CE (Slave Select, Chip Select o Chip Enable): linea attraverso cui il master può selezionare a quale slave inviare i dati.

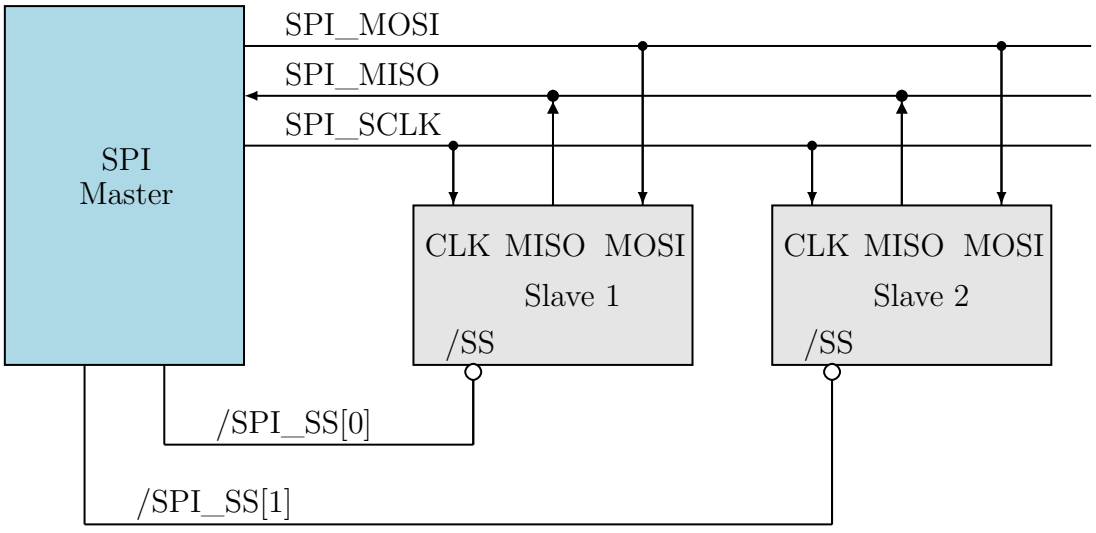


Figura 6.6: Schema del protocollo di comunicazione SPI.

Uno degli aspetti più rilevanti dei protocolli di comunicazione sincroni riguarda il clock. Il segnale di clock sincronizza l'uscita dei bit di dati dal master al campionamento dei bit da parte dello slave. Per ogni ciclo di clock viene trasferito 1 bit, quindi la velocità di trasferimento dei dati è determinata dalla frequenza di tale segnale.
L'avvio della comunicazione è effettuato dal master ed è quest'ultimo che configura e genera il segnale di clock. I controller SPI presenti su Raspberry Pi sono gestiti dal driver Linux spi_bcm2708 che supporta 15 diverse velocità, da 7.629 Hz fino ad un massimo di 125 MHz.

Per quanto riguarda la linea chiamata Slave Select, questa è necessaria al master per scegliere con quale slave esso intende comunicare. Tale intenzione viene esplicitata impostando la linea SS dello slave ad un livello logico basso. In uno stato inattivo, cioè in cui non vi è una trasmissione in atto, la linea viene mantenuta a un livello logico alto. Come anticipato nel precedente paragrafo, Raspberry Pi dispone di due master SPI: SPI0 che ha due linee SS e SPI1 che ne ha tre; per cui è possibile collegare in parallelo fino a 5 slave.

Il processo di trasmissione dei dati[8], rappresentato nella figura 6.7, si articola nei passi seguenti:

1. Il master invia il segnale di clock (figura 6.7a).
2. Il master attiva lo slave con cui comunicare ponendo la linea Slave Select a un livello logico basso (figura 6.7b).
3. Il master invia i dati allo slave un bit alla volta, usando la linea MOSI e inviando i dati a partire dal MSB (figura 6.7c). Lo slave legge i bit nell'ordine in cui vengono ricevuti.
4. Se è necessaria una risposta da parte dello slave, questo la invia un bit alla volta, a partire dal LSB, usando la linea MISO (figura 6.7d).

Per definire quando il dato presente sulle linee MISO e MOSI è valido si tiene conto della polarità e della fase del segnale di clock sulla linea SCLK.

Il driver bcm2708 dei microcomputer Raspberry Pi prevede per i suoi controllori SPI tre modalità di funzionamento: Standard, Bidirezionale e LoSSI (Low Speed Serial Interface). La prima di queste implementa il protocollo standard che utilizza quattro linee per la comunicazione. La modalità Bidirezionale invece, segue le stesse regole del protocollo standard ad eccezione della trasmissione dati che viene eseguita utilizzando un'unica linea (MOMI) anziché due (MISO e MOSI). In questa modalità la comunicazione diventa quindi half duplex e per lo scambio dei dati (pin MOMI) viene utilizzato il pin MOSI. La terza modalità di funzionamento, LoSSI, è uno standard comunemente usato per l'invio di comandi e il trasferimento di dati a controller LCD compatibili con MIPI DBI type C.

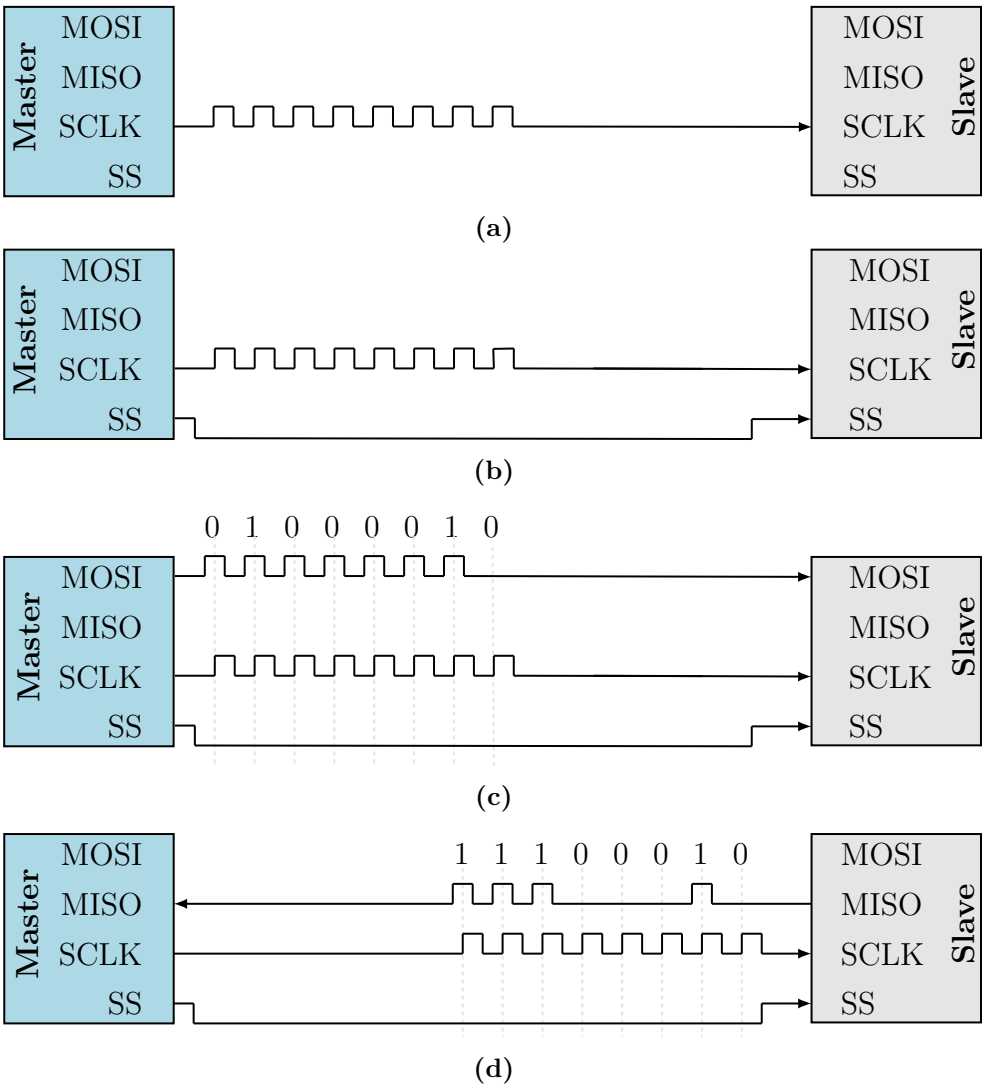


Figura 6.7: Processo di trasmissione dei dati nel protocollo SPI.

Il protocollo di comunicazione SPI ha le seguenti caratteristiche:

- Non necessita di bit di inizio e di fine trasmissione, per cui i dati sono inviati senza interruzioni.
- Non necessita di un sistema di indirizzamento come avviene per I^2C.
- La velocità di trasferimento è molto più alta di quanto ottenibile tramite I^2C.
- Permette di inviare e ricevere dati contemporaneamente, avendo due linee separate per tali funzioni (full duplex).
- Richiede l'utilizzo di tre (modalità bidirezionale) o quattro fili di collegamento

(modalità standard), pertanto necessita di altrettanti pin su un microcontrollore, a differenza di I^2C e UART che ne usano solo due.

- Non prevede acknowledgement, per cui non è possibile avere una conferma di ricezione dei dati, presente invece in I^2C.
- Non prevede forme di controllo degli errori nel flusso di dati, come nel caso del bit di parità in UART.

6.2.2 I^2C

I^2C combina alcune caratteristiche dei protocolli di comunicazione SPI e UART. Non solo è possibile avere molteplici dispositivi slave che rispondono a un singolo master ma è anche possibile avere più master che controllano un singolo slave.
Per la comunicazione vengono utilizzati due fili:

1. SDA (Serial Data): attraverso questa linea vengono inviati e ricevuti i dati dai dispositivi partecipanti nella connessione.
2. SCL (Serial Clock): fornisce il segnale di temporizzazione.

Le specifiche di questo protocollo sono state definite nel 1992 da Philips Semiconductors, diventata NXP Semiconductors nel 2006, e il protocollo è stato aggiornato più volte nel corso degli anni. Le specifiche attuali sono definite dalla versione 6 del documento UM10204, dell'aprile 2014 [9].

La velocità di trasmissione dei dati varia a seconda dell'implementazione del protocollo da parte del produttore di un dispositivo:

- Standard-mode: 100 Kbit/s.
- Fast-mode: 400 Kbit/s.
- Fast-mode Plus: 1 Mbit/s.
- High-speed mode: 3,4 Mbit/s.
- Ultra Fast-mode: 5 Mbit/s.

Il numero di dispositivi utilizzabili contemporaneamente da I^2C è molto alto e dipende dalle caratteristiche di indirizzamento da questi implementate. Lo standard prevede due tipi di indirizzamento: a 7 bit e a 10 bit. L'indirizzamento a 7 bit è il più diffuso e permette fino ad un massimo di 128 indirizzi diversi, inclusi alcuni indirizzi riservati, e ciò comporta la possibilità di utilizzare un numero piuttosto elevato di componenti e dispositivi attraverso lo stesso bus I^2C.

I dati vengono trasferiti sotto forma di messaggi, costituiti a loro volta da frame di dati lunghi 8 bit l'uno. Ciascun messaggio è costituito da un frame contenente

l'indirizzo dello slave e uno o più frame che contengono i dati da trasmettere. I messaggi contengono inoltre il comando di start e stop, il Read/Write Bit e l'ACK/NACK Bit. Il comando di start consiste nel porre a livello logico basso la linea SDA, prima di porre a livello basso anche la linea SCL, mentre il comando di stop consiste nel portare al livello logico alto il segnale SDA, successivamente alla stessa operazione sulla linea SCL. Il Read/Write Bit è un singolo bit posto a 0 se il master sta inviando dati ad uno slave (scrittura) e viceversa, a 1, quando il master sta richiedendo dei dati allo slave (lettura).
Ciascun frame di un messaggio è seguito da un bit di acknowledge/ no-acknowledge che fornisce l'indicazione di ricezione del frame da parte dello slave.

Il processo di trasmissione dei dati si articola nei seguenti passi[10], rappresentati in figura 6.8:

1. Il master invia il comando di start a ciascuno slave connesso, portando a livello logico basso la linea SDA, prima di porre a livello basso anche la linea SCL.
2. Il master invia a ciascuno slave l'indirizzo del dispositivo con cui esso vuole comunicare, seguito dal bit Read/Write.
3. Ogni slave confronta quindi l'indirizzo inviato dal master e se questo corrisponde al proprio, invia un bit ACK (pari a 0) al master. Se l'indirizzo non corrisponde, lo slave non esegue alcuna azione e la linea SDA rimane al livello logico alto.
4. Il master invia o riceve il frame di dati.
5. In seguito al trasferimento di ciascun frame di dati, il dispositivo destinatario invia un bit ACK (pari a 0) al mittente per notificarne la ricezione.
6. Per fermare il processo di trasmissione il master invia il comando di stop allo slave, ossia pone al livello logico alto il segnale SDA, successivamente alla stessa operazione effettuata sulla linea SCL.

A differenza del protocollo di comunicazione SPI, I^2C non è vincolato ad avere un solo master. Per ovviare al problema di più master che tentano di inviare o ricevere dati contemporaneamente sul bus I^2C, ogni master deve rilevare se la linea SDA è ad un livello logico basso o alto. Se la linea è a livello logico basso, allora un altro master ha il controllo del bus e si deve attendere che la linea torni ad un livello alto per trasmettere il proprio messaggio.

Sulla scheda Raspberry Pi Zero W il componente che gestisce il bus I^2C prende il nome BSC, acronimo di Broadcom Serial Controller, ed è conforme alla versione 2.1 del protocollo I^2C, rilasciata a gennaio 2000. La massima velocità raggiungibile è di 400 Kbit/s, corrispondente all'implementazione Fast-mode del bus.

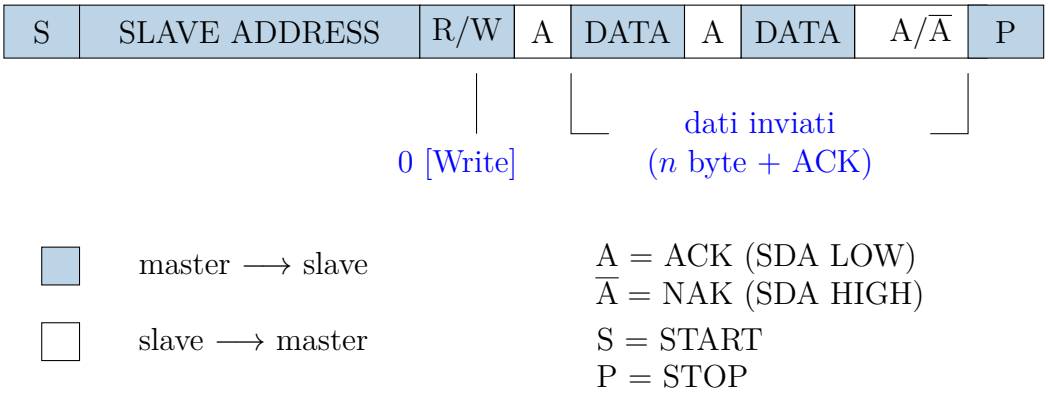


Figura 6.8: Sequenza di trasferimento dei dati dal master ad uno slave.

Il BSC consente di raggiungere una velocità massima di 400 Kbit/s (implementazione Fast-mode del bus), permette trasmissioni che coinvolgono un singolo master e supporta sia l'indirizzamento a 7 bit sia quello a 10 bit.

I vantaggi principali di questo protocollo di comunicazione sono senz'altro il minor numero di collegamenti fisici necessari rispetto a SPI e la conferma di ricezione dei dati da parte degli slave, cui si aggiunge il controllo di errori, tramite il bit di parità, nei frame di dati trasmessi. Alcuni svantaggi sono dati invece dalla velocità di trasmissione dei dati molto più lenta rispetto a SPI e dalla dimensione dei frame di dati, limitata a 8 bit.

6.3 Grafica

In questo paragrafo vengono trattati alcuni aspetti relativi alla dimensione visiva del progetto, riguardante i cubi. Ciascun dispositivo è dotato di uno schermo che occupa la faccia superiore dello stesso e attraverso il quale l'utente interagisce con lo spazio narrativo sviluppato sui cubi, in aggiunta alla manipolazione degli stessi nello spazio fisico.

Dal punto di vista dell'utente, il display agisce non solo come dispositivo di output ma anche come dispositivo di input, che tramite il touchscreen converte un'azione dell'utente in istruzioni che modificano lo stato del dispositivo fisico, nella fattispecie il cubo, producendo una reazione dello stesso. Questa reazione, all'interno del progetto, si sviluppa in forma sonora e in forma visiva. In questo paragrafo e nel paragrafo successivo, 6.4, verranno descritti i componenti che rendono possibile il feedback visivo all'utente.

Display

I display sono sempre più presenti nella nostra vita quotidiana, ne portiamo uno in tasca, lo smartphone, e sempre di più ne indossiamo uno, lo smartwatch. Infatti, secondo la società di analisi IHS, nel terzo trimestre del 2019 i produttori di display hanno consegnato 473 milioni di schermi per smartphone [11], 57 milioni di display per smartwatch e relativamente a quest'ultima tipologia la stima per il 2019 è pari ad un totale di 200 milioni di display consegnati, in netta crescita rispetto al 2018[12].

Il display svolge nella maggior parte dei casi un ruolo critico all'interno di un sistema in quanto ha il compito di presentare all'utente l'output o i risultati di un'elaborazione sotto forma di contenuti visivi ed è sottoposto al vincolo di eseguire tale compito in tempo reale. In questo senso si può parlare di interfaccia unidirezionale, un paradigma rimasto intatto per lungo tempo. Oggi però i dispositivi di visualizzazione quali i display non svolgono più solo una funzione di presentazione del risultato di un'elaborazione bensì essi diventano bidirezionali, agendo al contempo come dispositivi di input e di output[13]. Tali dispositivi diventano quindi sempre più interattivi, grazie all'integrazione di sensori e componenti che danno vita alle Natural User Interfaces (NUI), le quali permettono un'interazione diretta e in linea con il comportamento *naturale* dell'uomo, rappresentando inoltre l'evoluzione delle interfacce a riga di comando (CLI) e delle attuali GUI.

Ad ogni modo l'obiettivo di questo paragrafo è introdurre il principio di funzionamento dei display, intesi come dispositivi di output, descrivendo infine il tipo di display scelto nell'ambito del progetto di tesi.

Generalmente i display possono essere divisi in due macro categorie:

1. *Transmissive*: necessitano di una sorgente di luce per la visualizzazione di un'immagine e questa è integrata sul retro del pannello.
2. *Reflective*: viene sfruttata come sorgente luminosa la luce ambientale, riflessa opportunamente da uno specchio sottostante al pannello.

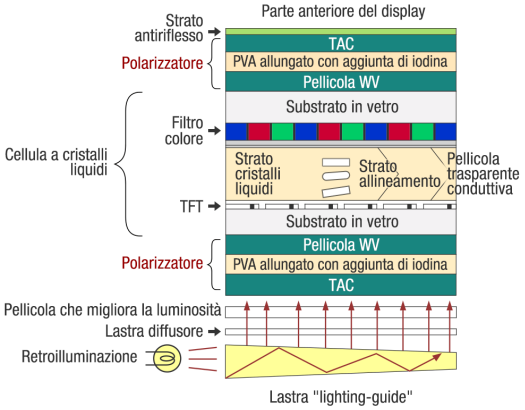
La maggior parte dei display ricadono nella prima tipologia e sono presenti in una vasta gamma di dispositivi di uso quotidiano, ma anche i display *reflective* riscontrano una diffusione crescente negli ultimi anni, anche grazie a dispositivi come gli eBook reader che implementano nella maggior parte dei casi uno schermo denominato e-Ink o e-Paper. Infine vi sono anche dispositivi di visualizzazione, *transflective*, che combinano entrambe le due tecnologie appena elencate.

I display a cristalli liquidi, LCD, sono attualmente la tipologia di visualizzazione più diffusa, essendo impiegata negli smartphone, smartwatch, computer, tv ecc. Nei display LCD tradizionali la sorgente della retroilluminazione era costituita da una

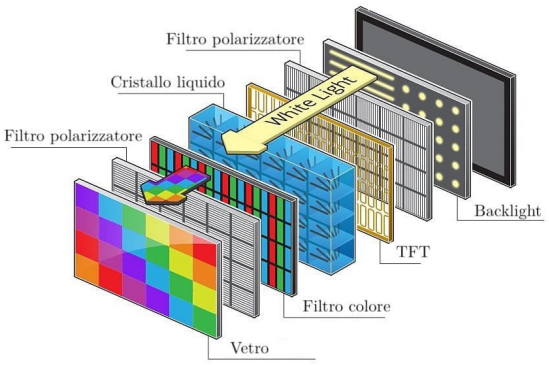
lampada fluorescente (CCFL) mentre ad oggi vengono utilizzati principalmente i LED.

Tali display sono basati sulle proprietà fisiche dei cristalli liquidi, ossia composti organici che al variare della temperatura mostrano una fase intermedia tra la fase liquida e quella solida. In questi cristalli liquidi le molecole, pur avendo la libertà di movimento, tendono ad assumere una posizione ed un orientamento. Le fasi con proprietà liquido cristalline, mesofasi, sono tre: (1) nematica, in cui le molecole tendono ad essere tutte orientate nella stessa direzione, (2) colesterica, in cui le molecole tendono ad essere di poco disallineate e (3) smettica, in cui oltre ad allinearsi, le molecole tendono a disporsi su piani paralleli. Gli LCD sfruttano opportunamente le caratteristiche ottiche delle mesofasi e utilizzano per lo più la mesofase nematica.

Dal punto di vista costruttivo, figura 6.9, il cristallo liquido si trova racchiuso tra due superfici solcate, in cui i solchi di una superficie, definiti direttori, sono perpendicolari rispetto ai solchi dell'altra. Agli estremi del cristallo liquido vi sono inoltre due filtri polarizzatori sfasati di 90°.



(a) Fonte: FUJIFILM.



(b) Fonte: AOI Display.

Figura 6.9: Dettaglio di un pannello LCD TFT.

La luce proveniente dalla sorgente luminosa attraversa il primo filtro, viene quindi polarizzata e modifica la propria polarizzazione seguendo l'orientamento delle molecole; transitando per il cristallo liquido viene ruotata di 90° e infine attraversa il secondo filtro polarizzatore [14].
Applicando un campo elettrico le molecole del cristallo liquido tendono a disporsi nella direzione del campo, pertanto in presenza di una tensione la luce viene polarizzata dal primo filtro, transita per il cristallo liquido ma viene bloccata dal secondo filtro, mentre in assenza di tensione la luce passa correttamente. Tramite

un campo elettrico si può quindi controllare la quantità di luce che attraversa il cristallo liquido.

I display a cristalli liquidi sono costituiti da matrici di celle. Per display di dimensioni piccole ogni cella è controllata da un elettrodo, mentre per display sufficientemente grandi si utilizza un reticolo di elettrodi trasparenti in modo da attivare ogni pixel con un meccanismo del tipo riga-colonna. Si parla in questo caso di display LCD a matrice passiva. In essi quindi si trova un gruppo di contatti per ogni riga e colonna dello schermo e si può controllare un solo pixel alla volta mentre gli altri devono ricordare il proprio stato finché il circuito di controllo non si dedica nuovamente a essi. La risposta di questi LCD è molto lenta e non sono in grado di seguire i cambiamenti veloci nel contenuto dell'immagine, oltre ad essere affetti dal fenomeno di *ghosting* (i pixel accesi interferiscono con quelli spenti appartenenti alle stesse righe e colonne)[15].

Nei display a matrice attiva o TFT (Thin Film Transistor), ad ogni elemento di ciascun pixel è associato un transistor, permettendo quindi di migliorare i tempi di risposta e il rapporto di contrasto nonché la luminanza del display. Inoltre, potendo controllare con precisione l'accensione dei singoli pixel, non è presente l'effetto ghosting.

La visualizzazione di un'immagine a colori sul display è resa possibile dalla divisione di ogni pixel in tre sub-pixel, relativi alle componenti rosso, verde e blu (RGB), e tramite la presenza di un filtro colore, quale ad esempio lo schema Bayer.

Per quanto riguarda la scelta del display utilizzato, i criteri di valutazione sono stati dettati innanzitutto dalle dimensioni del dispositivo di gioco che, come spiegato nel capitolo 5, sono di $88 \times 88 \times 88$ mm; per cui le dimensioni della breakout board del display sarebbero dovute essere inferiori alla faccia del cubo. Strettamente collegato a tale aspetto vi è anche l'aspect ratio del display, in quanto idealmente il display avrebbe dovuto essere quadrato, ottimizzando in questo modo la superficie adoperabile sulla faccia del cubo e contribuendo così ad una maggiore continuità visiva e ad un'esperienza più immersiva quando si utilizzano più di due cubi affiancati.
Il secondo criterio è stato quello di compatibilità con il Raspberry Pi, per cui la possibile scelta del display si è notevolmente ristretta a pochi display che usano i driver supportati da Raspberry Pi. Correlato a questo aspetto vi è infine anche l'interfaccia di comunicazione che avrebbe dovuto essere necessariamente seriale al fine di lasciare libero sul Raspberry Pi un numero di pin sufficiente per il collegamento degli altri componenti elettronici.
Infine, in considerazione dell'attuale fase di prototipazione in cui si collocano i cubi, oltre all'aspetto economico, l'ultimo parametro di valutazione è stato dato dalle specifiche tecniche dei display soddisfacenti i due requisiti sopra citati e quindi non scartati. La scelta effettuata sarà descritta nel sottoparagrafo successivo.

LED

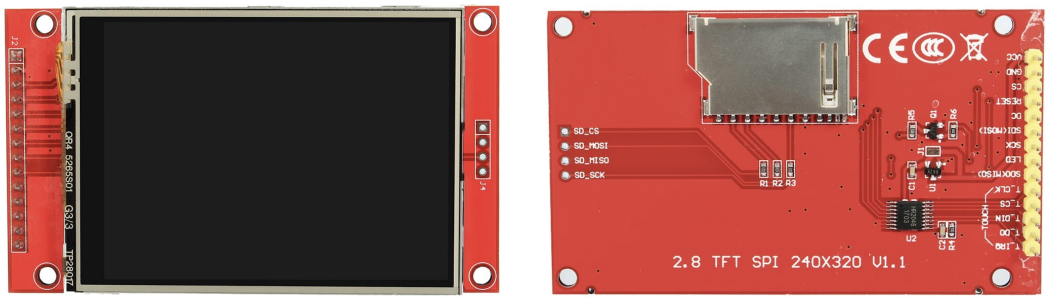
In aggiunta al display, ciascun cubo è dotato di una serie di LED che variano il proprio colore e la luminosità a seconda delle azioni effettuate dall'utente.

La tipologia di LED utilizzata, a differenza del LED *tradizionale*, racchiude nello stesso contenitore, di dimensioni estremamente ridotte, tre diversi diodi a emissione di luce: uno rosso, uno verde ed uno blu. Il principio che permette la percezione di un colore uniforme, e non le componenti RGB separate, è lo stesso che trova applicazione nei display, in cui ogni pixel è composto da tre sub-pixel, pari alle componenti RGB. I colori dei diodi vengono quindi miscelati con intensità diverse per produrre una grande varietà di colori.
Le caratteristiche e il funzionamento del LED verranno approfondite nel sottoparagrafo 6.3.2.

6.3.1 Display LCD 2.8" ILI9341

Un componente chiave di CuBee è senz'altro il display dotato di un pannello touchscreen di tipo resistivo. Si tratta di uno schermo LCD TFT generico, ovvero non prodotto da un marchio in particolare, ed è basato sul driver Ilitek ILI9341 utilizzato nella gestione del display, mentre il touchscreen è affidato al driver Texas Instruments ADS7846, trattato nel paragrafo 6.4.

La gestione di un display è un'attività impegnativa per un elaboratore ed anche per questo si è scelto di utilizzare un Raspberry Pi e non un semplice microcontrollore come Arduino o ESP. Oltre all'allocazione di memoria necessaria per i dati relativi ai contenuti visivi è necessaria anche una sufficiente larghezza di banda, che può diventare significativa per i display ad alta risoluzione [16].



(a) **(b)**

Figura 6.10: Dettaglio del display LCD TFT utilizzato. Fonte: Banggood.

Il display utilizzato, mostrato in figura 6.10, ha una risoluzione QVGA, ossia 320 × 240 pixel, per cui sul pannello sono presenti 76.800 pixel e, con una profondità

di colore pari a 18 bit/pixel, la memoria minima richiesta per ogni frame è pari a 172,8 kB. Tale spazio richiesto coincide con quello disponibile nella memoria GRAM (Graphical Random Access Memory) presente nel driver del display. Sul Raspberry Pi sono presenti due framebuffer, ossia dispositivi virtuali i quali, come la maggior parte dei dispositivi su un sistema Linux, possono essere aperti e gestiti come un normale file, e su questi file sono possibili le operazioni di lettura e scrittura al fine di generare un'interfaccia grafica. I framebuffer presenti sono i seguenti:

1. /dev/fb0 che è assegnato all'uscita del connettore HDMI presente sulla scheda e permette di sfruttare l'accelerazione hardware per l'uscita HDMI.
2. /dev/fb1 che è assegnato invece ai display che utilizzano l'interfaccia SPI.

A seconda della frequenza di aggiornamento del display il flusso di dati può essere piuttosto impegnativo in termini di trasferimento dei dati: per il display utilizzato nel progetto, quindi con una risoluzione pari a 320×240 pixel, una profondità di colore pari a 18 bit/pixel e una frequenza di aggiornamento uguale a 30 fps, i dati che devono essere trasmessi dal Raspberry Pi al (driver del) display ammontano a 5,184 MB. Di conseguenza non è possibile utilizzare l'interfaccia SPI con elevate frequenze di aggiornamento o per il collegamento di display ad alta risoluzione.

Le dimensioni del PCB su cui si trova il display sono pari a 86×50 mm, che sono di poco inferiori alle dimensioni del cubo. Al fine di realizzare una struttura del dispositivo solida si è limato il display tagliando una parte della scheda sulla quale si trovano i pin per il collegamento ad una scheda SD, il cui alloggiamento è presente sul retro del display e che è stato dissaldato per ridurre lo spazio occupato dal display all'interno del cubo.

Per quanto riguarda invece le dimensioni dell'area attiva del display, questa è pari a 65×46 mm ed essa è stata ulteriormente ridotta in considerazione del fatto che uno dei requisiti del display era quello di avere un aspect ratio $1 : 1$. Per cui l'area attiva effettivamente visibile sul cubo è di 46×46 mm, all'interno di una faccia di 88×88 mm, ossia il display occupa il 27,33% della faccia del cubo. Tale valore, seppur non ottimale, rappresenta il miglior risultato a cui è stato possibile arrivare in questa fase di prototipazione.

Nella tabella 6.1 sono riepilogate le caratteristiche principali del display. Lo schema di collegamento al Raspberry Pi è invece riportato nella figura 6.11. Nello stesso schema sono presenti anche le linee di comunicazione del controller touchscreen.

Tabella 6.1: Caratteristiche principali del display LCD TFT utilizzato.

Dimensioni schermo	2,8″
Dimensioni PCB	86 × 50 mm
Area attiva ($L \times H$)	65 × 46 mm
Risoluzione ($L \times H$)	320 × 240 px
Tecnologia	a-Si TFT a matrice attiva
Driver	ILI9341
Interfaccia di collegamento	Seriale (SPI), Parallela (6-bit RGB, 16-bit RGB, 18-bit RGB)
Profondità di colore	16 bpp o 18 bpp
Disposizione sub-pixel	Stripe
Graphics RAM/ Framebuffer	172,8 kB
Tensione di alimentazione	3,3 – 5 V
Retroilluminazione	LED

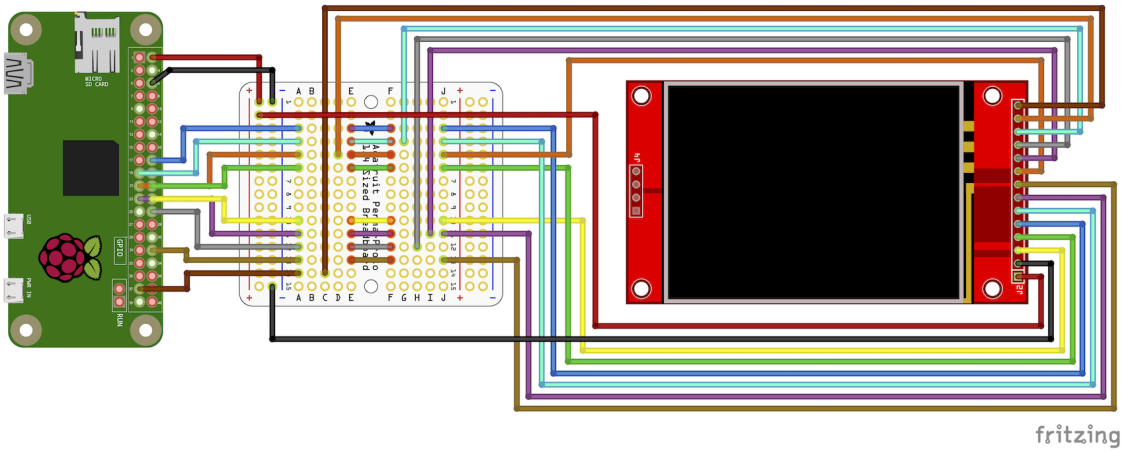


Figura 6.11: Schema di collegamento del display LCD al Raspberry Pi. Realizzato con Fritzing.

6.3.2 LED SMD5050 WS2812

Altro componente legato all'aspetto visivo dei cubi sono i moduli LED RGB di tipo indirizzabile e dotati del circuito di controllo WS2812 della Worldsemi. Un singolo LED, mostrato in figura 6.12, ha dimensioni pari a 5 × 5 mm ed è costituito da

tre diodi, figura 6.13, indicati anche come pixel: uno rosso, uno verde ed uno blu. Ciascun LED contiene inoltre un oscillatore interno ($f = 800$) kHz, un flip-flop e un ripetitore digitale che amplifica il segnale e rimodella gli impulsi trasmessi da un LED a quello successivo.

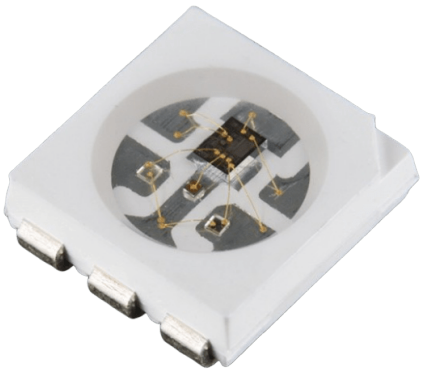


Figura 6.12: Dettaglio LED RGB. Fonte: Sparkfun.

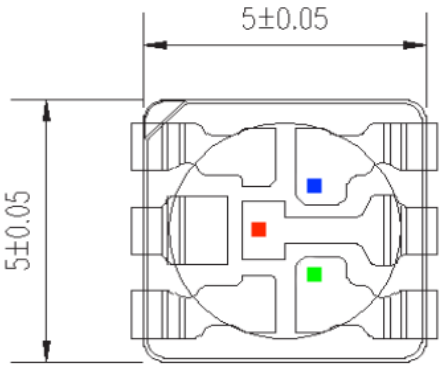


Figura 6.13: Schema LED RGB. Fonte: Kento.

In ogni cubo sono presenti 4 LED, uno per ogni faccia laterale del cubo e questi sono collegati in cascata. Il colore di ciascuno di questi LED può essere controllato tramite un'interfaccia seriale asincrona che utilizza frame di 24 bit, ossia 8 bit per ogni pixel. Ciascun pixel ha quindi 256 livelli di luminosità possibili e complessivamente ogni LED può generare fino a 16.777.216 colori diversi.
A livello di generazione del segnale si verifica che in un sistema con n LED devono essere generati n frame di dati, ovvero $n \times 24$ bit ad ogni ciclo di aggiornamento del sistema.

Questi bit vengono codificati nel formato GRB, ossia il bit iniziale ricevuto da un LED è costituito dal bit meno significativo della luminosità del pixel blu mentre l'ultimo bit è costituito dal MSB della luminosità del pixel rosso.

Successivamente al reset iniziale dei pixel, il primo LED della catena riceve tramite il pin DIN i dati dal Raspberry Pi, trattiene i dati che lo riguardano direttamente, ovvero i primi 24 bit, mentre il flusso di dati rimanente viene trasmesso al LED successivo.

L'interfaccia seriale utilizza una codifica che rappresenta sia il livello logico alto, 1, sia il livello logico basso, 0, con un segnale a onda quadra. La differenza tra i due livelli è data dalla durata maggiore del livello alto del segnale, figura 6.14, e nello specifico si ha che uno 0 è caratterizzato da $T_{0H} = 0{,}35\,\mu s$ mentre un 1 è definito da $T_{1H} = 2 \times T_{0H} = 0{,}7\,\mu s$. Il tempo di trasmissione di un bit dichiarato dal produttore del controller WS2812 è $T_{TX,bit} = 1{,}25 \pm 0{,}6\,\mu s$.

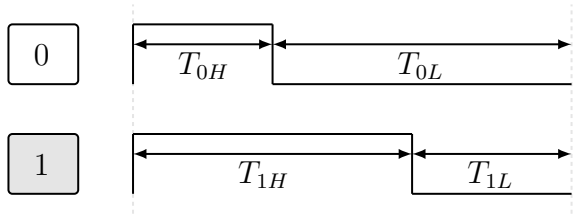


Figura 6.14: Codifica per la trasmissione dei dati implementata dal controller WS2812.

Per segnalare la fine del frame, e quindi indicare ai LED di aggiornare il proprio colore secondo le istruzioni ricevute, la linea di comunicazione viene mantenuta ad un livello di tensione basso per un tempo T_{reset} di almeno 50 µs.

Nel caso di un sistema con n LED, per ogni ciclo di aggiornamento completo del sistema è richiesto un tempo di trasmissione $T_{TX,cycle}$ pari a:

$$T_{TX,cycle} = n \times (24 \times T_{TX,bit} + T_{reset}) \tag{6.1}$$

A partire dal tempo di trasmissione ricavato tramite la 6.1, trascurando i tempi di trasmissione del segnale di controllo dal dispositivo di elaborazione ed eventuali ritardi, è possibile ricavare la frequenza massima di aggiornamento della catena di LED, che nel caso di un sistema contenente 4 LED è pari a circa 3,078 kHz. Collegando fino a 1024 LED tramite una singola linea di comunicazione è possibile ottenere una frequenza di aggiornamento di circa 30 Hz.

Infine per quanto riguarda l'alimentazione ciascun pixel assorbe fino a 20 mA e il LED richiede una tensione di 5 V. Lo schema di collegamento dei LED al Raspberry Pi è riportato nella figura 6.15.

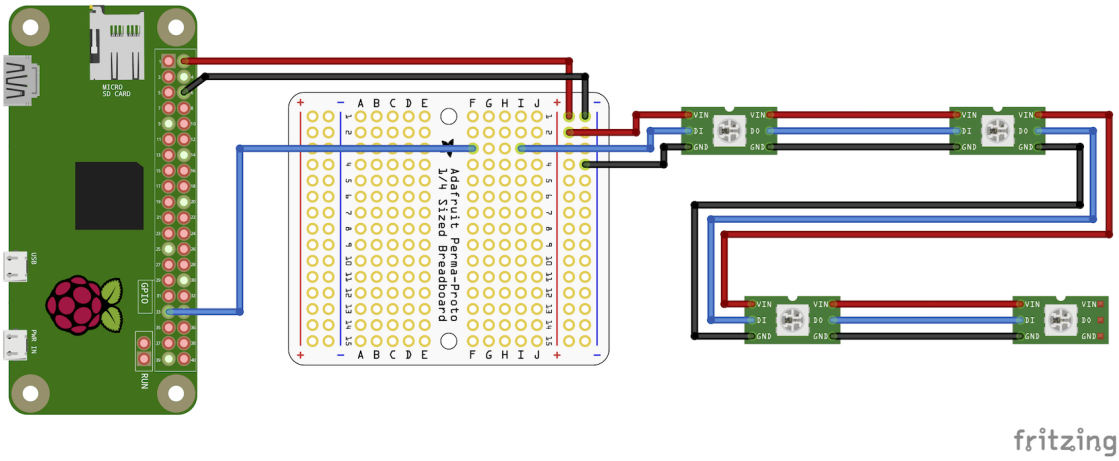


Figura 6.15: Schema di collegamento dei LED al Raspberry Pi. Realizzato con Fritzing.

6.4 Interazione: Touch

Come accennato nel paragrafo precedente, l'interazione con un dispositivo fisico, quale ad esempio uno smartphone, un computer, o il cubo nel caso di questo progetto, è possibile tramite dispositivi di input che convertono un'azione dell'utente in istruzioni che modificano lo stato del dispositivo in oggetto, producendo una reazione dello stesso.
Gli interruttori ed i pulsanti rappresentano sicuramente uno dei metodi più semplici e diretti a tal scopo, tuttavia ad essi è associata anche una dimensione fisica che diventa problematica ad esempio per quanto riguarda i dispositivi mobili.
Questo aspetto richiede quindi un'attenta valutazione da parte dei progettisti nella scelta di quali pulsanti fisici dotare un dispositivo portatile. A tal fine è necessario uno studio sulle funzioni ritenute più importanti e/o maggiormente utilizzate dagli utenti, in modo da valutare quali di queste devono essere implementate nei dispositivi e agevolare quindi l'utente durante l'utilizzo del dispositivo.

Ad esempio l'attivazione o la disattivazione della suoneria di un dispositivo mobile fino ai primi anni del 2000, era gestita nella maggior parte dei casi a livello software e richiedeva all'utente di navigare tra una serie di voci del menu, quindi non era un'operazione immediata; oppure in alcuni telefoni si poteva eventualmente attivare tale funzionalità tenendo premuto un pulsante della tastiera fisica, ad esempio il pulsante cancelletto per i telefoni Nokia. Entrambe queste soluzioni non erano ottimali dal punto di vista della User Experience e con l'introduzione dell'iPhone Edge, nel 2007, questa funzionalità, reputata importante dai progettisti di Apple, ha trovato posto in un pulsante switch sul lato sinistro del dispositivo.

A partire dal 2007 la maggior parte dei tasti e pulsanti fisici presenti sui dispositivi mobili sono stati progressivamente abbandonati dai produttori di smartphone in favore di un nuovo dispositivo di input: il touchscreen.

Il concetto di touchscreen fu teorizzato già nel lontano 1962 da un ingegnere della Royal Radar Establishment in Inghilterra, Eric Arthur Johnson, che ne parlò in un articolo di sole due pagine dal titolo *Touch Display—A novel input/output device for computers*[17]. Il touchscreen, di tipo capacitivo, sarebbe stato successivamente implementato al CERN, nel 1973, dai due ingegneri Bent Stumpe e Frank Beck.

Per via della mancanza di parti mobili, i touchscreen si prestano molto ad applicazioni destinate a luoghi pubblici, trovando ampio uso ad esempio negli aeroporti e sempre di più nei musei. La facilità d'uso e la semplicità rappresentano alcuni degli aspetti che hanno contribuito alla diffusione di questa tecnologia, che tuttavia pone ulteriormente l'accento sull'importanza dell'interfaccia grafica sottostante, vitale per l'usabilità di un sistema.

Per quanto riguarda i cubi del nostro progetto, in base al contesto per cui il dispositivo è stato progettato, non sono previsti pulsanti accessibili agli utenti. Tale scelta è stata dettata dalle caratteristiche stesse dei cubi, tra cui la modularità, che non hanno lasciato spazio a questa tipologia di componenti, ma anche dalla mancanza di eventuali funzioni necessarie durante l'utilizzo dei dispositivi e non implementabili altrimenti.
Vi è tuttavia un pulsante di accensione e spegnimento del dispositivo, posizionato sulla base del cubo ma è adoperabile esclusivamente tramite uno spillo e riservato al personale incaricato della manutenzione del museo. Questa scelta è stata effettuata al fine di evitare che gli utenti del museo possano accidentalmente spegnere i cubi.

Tipologie di touchscreen

Le principali tipologie di touchscreen sono due: il touchscreen resistivo e quello capacitivo. Sebbene quest'ultimo sia quello più diffuso e presente su tutti i dispositivi mobili di nuova generazione, il touchscreen resistivo è ancora presente in dispositivi di tipo industriale e in dispositivi consumer più economici.

Il modo più semplice di realizzare un touchscreen capacitivo consiste nel rivestire la superficie dello schermo con uno strato trasparente di materiale conduttore, generalmente ossido di indio-stagno. Applicando una tensione ai quattro lati dello schermo si crea un campo elettrico uniforme su tutta la sua superficie.
Quando si tocca lo schermo con il dito, o con un qualsiasi oggetto ricoperto di un materiale conduttore, avviene una distorsione del flusso di elettroni e la matrice di condensatori a film posizionata sotto il pannello touchscreen permette di rilevare e localizzare la variazione della capacità elettrica. Tramite il controllore del touchscreen è possibile individuare in modo rapido e preciso il punto in cui è avvenuto il tocco e inviare tale informazione al componente o dispositivo incaricato dell'elaborazione.

Tra i vantaggi del touchscreen capacitivo vi è il multi-touch, che consente quindi di effettuare lo zoom, la rotazione di immagini e testi, nonché modalità di gioco impossibili con il tipo resistivo. Inoltre tale tipologia è generalmente più veloce e più sensibile al tocco.

Il funzionamento del touchscreen resistivo invece si basa sul rilevamento della pressione esercitata sullo schermo, tramite il dito o qualsiasi altro oggetto. Le due architetture più diffuse sono: 4-Wire e 5-Wire. All'interno di questo progetto è stata utilizzata la prima tipologia, 4-Wire e il funzionamento e le immagini che seguono fanno riferimento a tale architettura.

Un pannello touchscreen resistivo è composto da due strati resistivi trasparenti:

1. Uno strato superiore in materiale flessibile, tipicamente polietilene.

2. Uno strato inferiore di vetro.

Questi due strati, entrambi trasparenti, sono rivestiti con dell'ossido di indio-stagno (comunemente indicato con la sigla ITO), figura 6.16, e sono separati da alcuni distanziatori (*spacer* e *spacer dots*) che hanno il compito di far ritornare lo strato superiore nella posizione originaria in seguito ad una pressione, ma anche di evitare false rilevazioni causate dalla flessibilità del materiale in cui lo strato superiore è realizzato.

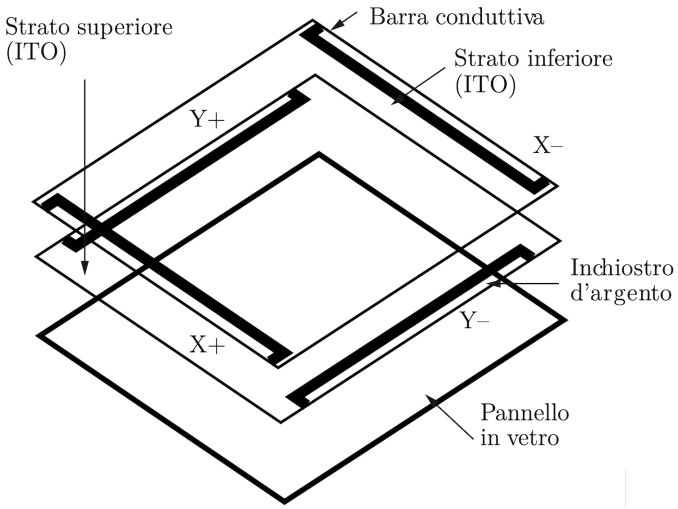


Figura 6.16: Touchscreen resistivo 4-Wire. Fonte: Texas Instruments.

Quando lo strato superiore viene premuto, le superfici resistive che ricoprono i due strati del pannello entrano in contatto e il controller del touchscreen può di conseguenza localizzare il punto in cui è avvenuta tale azione. Il contatto elettrico tra le due superfici resistive genera lo schema di resistenze mostrato nella figura 6.17 e siccome il materiale delle superfici è omogeneo, il valore delle resistenze R_{X+}, R_{X-}, R_{Y+} e R_{Y-}, è proporzionale alle coordinate (x, y) del punto di contatto. Per rilevare la coordinata x del punto interessato dal tocco si applica una tensione di polarizzazione nota ai capi dello strato X e si misura la tensione all'elettrodo Y+ a circuito aperto, lasciando non connesso l'elettrodo Y−. In questo modo è possibile determinare il rapporto delle resistenze R_X che costituiscono il partitore di tensione sul piano X e quindi determinare la coordinata x del punto interessato dal tocco[18]. Questo principio di funzionamento, del tutto analogo per la coordinata y, è illustrato nella figura 6.17.

Per la fase di prototipazione dei dispositivi oggetto di questa tesi è stato utilizzato un touchscreen resistivo 4-wire, di cui si proseguirà la trattazione, introducendo il controller ADS7846, nel sottoparagrafo successivo.

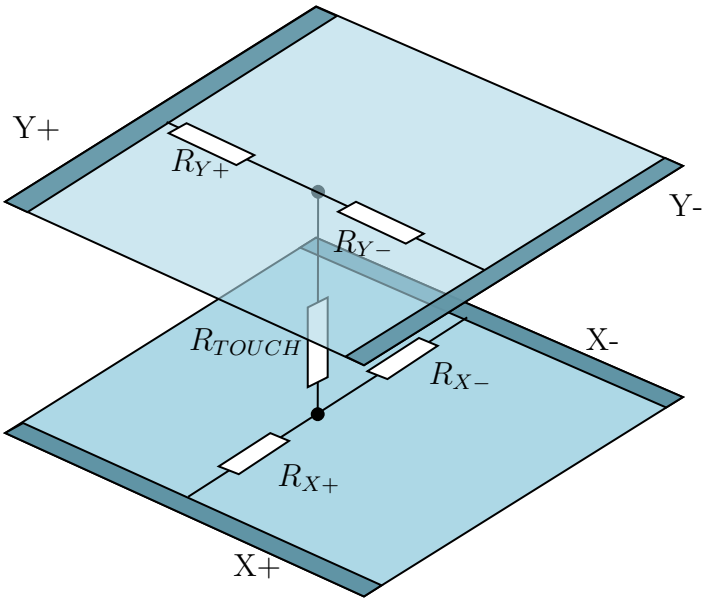


Figura 6.17: Principio di funzionamento di un touchscreen resistivo.

6.4.1 Modulo Texas Instruments ADS7846

Il touch screen controller, in seguito indicato con la sigla TSC, è il dispositivo che permette la comunicazione tra un pannello touchscreen e un dispositivo di elaborazione. Il TSC utilizzato in questo progetto è l'ADS7846 di Texas Instruments. Esso rientra nella tipologia di controller basata su tecnologia *command-based*, ossia il controller funziona esclusivamente sotto il controllo del dispositivo di elaborazione, definito anche host. Questo paragrafo si basa in larga parte sul materiale riportato nel datasheet del TSC utilizzato [19] e nelle note applicative fornite da Texas Instruments [20][21][22].

Sul display descritto precedentemente trova applicazione un touchscreen di tipo resistivo che utilizza il controller ADS7846 di Texas Instruments.

Come mostrato in figura 6.18, un controller touchscreen dispone di due interfacce:

1. L'interfaccia analogica, dalla quale transitano i dati dal pannello touchscreen al controller.
2. L'interfaccia digitale, dalla quale transitano i dati dal TSC al dispositivo di elaborazione.

Quest'ultima interfaccia permette il collegamento e lo scambio di dati con l'host tramite un bus SPI o I^2C. L'ADS7846 utilizza il bus SPI per comunicare i dati relativi alle coordinate (x, y) del punto di pressione, agendo come slave e quindi

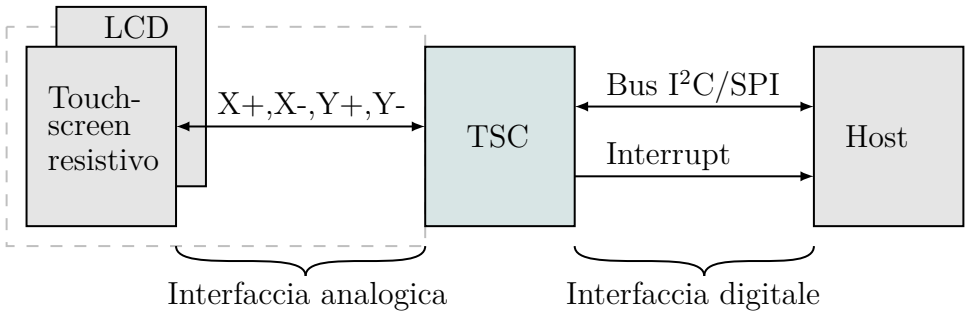


Figura 6.18: Schema di un sistema contenente un touchscreen 4-wire

comunicando i dati all'host unicamente quando il TSC rileva una pressione sul pannello touchscreen.

Generalmente i TSC command-based richiedono la completa attenzione dell'host durante il rilevamento di una pressione sul touchscreen. La sequenza delle operazioni eseguite dall'ADS7846, nonché dei dispositivi TSC command-based, è la seguente:

1. In assenza di pressione sul pannello touchscreen il controller viene spento e non vi sono dati in transito né sull'interfaccia analogica né su quella digitale.
2. Quando si ha una pressione sul pannello si verificano i seguenti step:
   (a) Il TSC rileva la pressione e invia all'host un interrupt (PENIRQ), sollecitandone l'immediata attenzione.
   (b) L'host riceve l'interrupt e avvia la corrispondente Interrupt Service Routine (ISR).
   (c) L'ISR invia un comando al TSC tramite il bus SPI (o I^2C per controller diversi dall'ADS7846) e richiede al TSC di leggere una coordinata del punto di tocco, generalmente la coordinata x.
   (d) Alla ricezione del comando indicato al punto (c), il TSC accende il driver relativo alla coordinata x.
   (e) Il TSC acquisisce il segnale relativo alla coordinata x e converte il segnale analogico in digitale, inviando i dati all'host un bit alla volta.
   (f) L'host riceve un'unità di dati relativi all'asse x, formata da 8 o 12 bit a seconda della configurazione del TSC. Se sono necessarie più unità di dati (ai fini di filtraggio o media) vengono ripetuti i passi (c) - (e).
   (g) L'ISR invia un comando al TSC tramite il bus SPI (o I^2C per altri controller) e richiede al controller di leggere un'altra coordinata del punto di tocco, generalmente la coordinata y.

(h) Alla ricezione del comando precedente, il TSC spegne il driver relativo alla coordinata x e accende il driver relativo a y.

(i) Il TSC acquisisce il segnale relativo alla coordinata y e converte il segnale analogico in digitale, inviando i dati all'host un bit alla volta.

(j) L'host riceve un'unità di dati relativi all'asse y. Se sono necessarie più unità di dati vengono ripetuti i passi (g) - (i).

(k) Alla fine della ricezione del set completo di coordinate (x, y) l'host abilita nuovamente PENIRQ e verifica se il pannello è ancora premuto. In caso affermativo ritorna al passo (b) per il prossimo set di coordinate.

3. Se non vi è più pressione sul pannello si ritorna allo step (1).

Nella figura 6.19 è rappresentato lo schema del funzionamento di base del controller utilizzato. In aggiunta alla coppia (x, y), l'ADS7846 fornisce due coordinate chiamate z_1 e z_2, le quali permettono di ricavare l'informazione sul livello di pressione nel punto di tocco, utile ad esempio per implementare azioni diverse all'interno dei giochi sviluppati e descritti nei capitoli successivi.

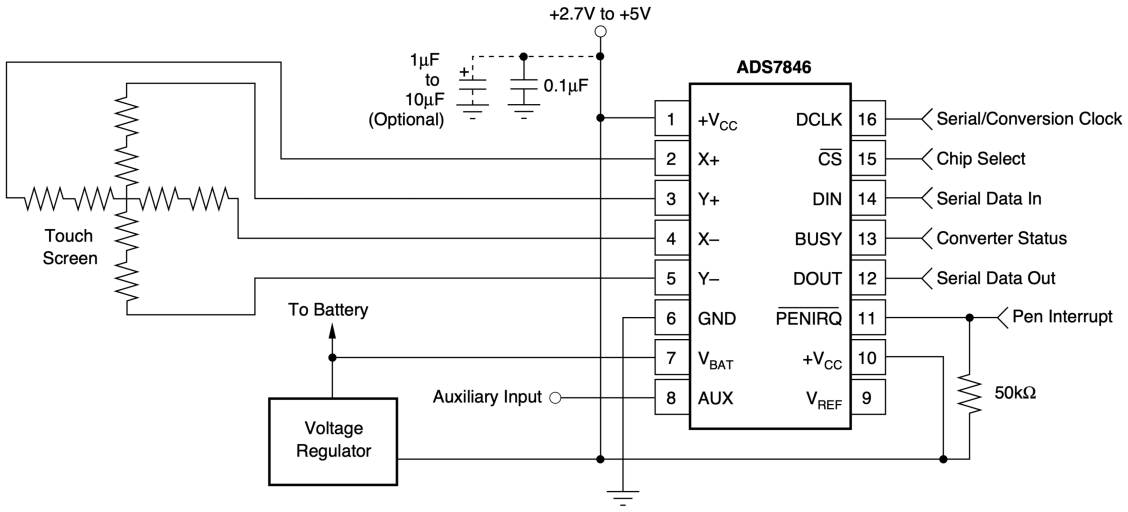


Figura 6.19: Schema operativo di base dell'ADS7846. Fonte: Texas Instruments.

Quando si utilizza un touchscreen come *human interface*, tipicamente ci si aspetta da 100 a 500 set di campioni validi al secondo e un controller touchscreen *command-based* potrebbe non essere in grado di fornirne un numero sufficiente. Per un dispositivo TSC che implementa il bus SPI, come l'ADS7846, anche se la velocità del bus generalmente non influenza il parametro SSPS (sample sets per second), a differenza di quanto può accadere nel caso del bus I^2C, vi possono essere complicazioni dovute al traffico intenso sul bus che può causare problemi sull'host e/o consumare più corrente.

La velocità di trasmissione dei dati diventa un aspetto sensibile soprattutto quando vengono utilizzate tecniche di filtraggio o viene effettuata una media dei valori letti, che comporta più ripetizioni della sequenza di operazioni vista precedentemente.

Per quanto riguarda l'interfaccia analogica, nel caso dei controller touchscreen command-based, l'intervallo tra un campione e l'altro è determinato dalla velocità di risposta dell'host e dalla sua larghezza di banda. Inoltre, in presenza di rumore nel TSC, l'host deve effettuare un'operazione di media dei dati o un opportuno filtraggio, operazioni che richiedono almeno due campioni per ciascuna coordinata del sistema, ed anche in questo caso il numero di campioni è determinato dalla larghezza di banda dell'host e dalle sue risorse.

Le caratteristiche principali dell'ADS7846 sono le seguenti:

- risoluzione di 12 bit;
- tempo massimo di conversione di un campione pari a 12 cicli di clock;
- tempo minimo di acquisizione di un campione pari a 3 cicli di clock;
- throughput rate massimo di 125 kHz;
- tensione di alimentazione compresa tra 2,7 V e 5,25 V;
- consumo inferiore a 0,75 mW (typ 2,7 V).

La maggior parte degli errori che possono influire sull'accuratezza e l'affidabilità del touchscreen può essere attribuita a tre fonti. La prima riguarda il rumore elettrico proveniente dal display, dalla retroilluminazione, dall'interazione con l'utente, dalle vibrazioni della superficie del pannello o dalle scariche elettrostatiche e dagli impulsi elettromagnetici causati dagli utenti e dall'ambiente. Le restanti fonti riguardano invece i fattori di ridimensionamento e i disallineamenti meccanici che hanno origine nell'assemblaggio del touchscreen con il display.
In genere, il TSC e il display in un sistema non hanno la stessa risoluzione, pertanto è necessaria un'operazione di ridimensionamento, effettuata attraverso l'introduzione di fattori di ridimensionamento. Ad esempio, in questo progetto il touchscreen si trova su un display LCD con una risoluzione di 320 pixel (x) $\times$ 240 pixel (y) mentre il controller ADS7846 ha una risoluzione a 12 bit, ovvero 4096 $\times$ 4096. I fattori di ridimensionamento per abbinare i dispositivi sono $k_X = S_X/S_{X'} = 320/4096 = 0{,}078125$ per la coordinata dell'asse x e $k_Y = S_Y/S_{Y'} = 240/4096 = 0{,}05859375$ per la coordinata dell'asse y, con S_X risoluzione dell'asse x dell'LCD, $S_{X'}$ risoluzione dell'asse x del controller e analogamente S_Y e $S_{Y'}$.
In una situazione in cui $k_X = 1$ e $k_Y = 0{,}9$, un punto $P(x', y') = (2, 2{,}22222)$ sul touchscreen dovrebbe essere fatto coincidere con il punto del display LCD di coordinate $(y, y) = (2, 2)$. Pertanto, la coordinata x di un controller touchscreen, x',

dovrebbe coincidere con il valore dell'LCD (host) $x = k_X \times x'$ e analogamente per quanto riguarda la coordinata y.

L'operazione di riduzione o eliminazione di eventuali incongruenze prende il nome di calibrazione ed essa permette di tradurre le coordinate riportate dal controller del touchscreen in coordinate che rappresentano con precisione il punto e la posizione dell'immagine sul display sottostante.
Il risultato della calibrazione, nel caso del Raspberry Pi, consiste in un insieme di valori che opportunamente utilizzati dal driver del controller ADS7846, permettono di far coincidere la superficie del pannello touchscreen con quella effettiva del display sottostante, consentendone quindi un utilizzo naturale da parte dell'utente.

Per quanto riguarda il collegamento del controller al Raspberry Pi è stato utilizzato il bus SPI0, condiviso con il display LCD che si avvale del pin CE0. Come previsto dal protocollo di comunicazione SPI, i pin adoperati sono quattro: SPI0 MOSI, SPI0 MISO, SPI0 SCLK e SPI0 CE1. A questi si aggiungono i pin di alimentazione, condivisi con il display in quanto entrambi i componenti sono saldati sullo stesso PCB, e il pin relativo al segnale di interrupt, PENIRQ.

6.5 Interazione: Motion sensing

Rilevare il movimento di un dispositivo all'interno di un determinato ambiente fisico e analizzare i dati forniti da una serie di sensori al suo interno è possibile grazie a dispositivi elettronici definiti *Inertial Measurement Unit*. Tali dispositivi dispongono, sullo stesso integrato, di due o più sensori: accelerometro, giroscopio, sensore di pressione, sensore di temperatura, magnetometro ecc. L'attività di rilevazione dei movimenti del sensore prende anche il nome di *motion sensing* o *motion tracking*.

I sensori che si trovano su una IMU sono generalmente realizzati con la tecnologia dei Micro-Electro-Mechanical Systems, abbreviati con l'acronimo MEMS, che permette di realizzare dispositivi microelettronici che integrano su uno stesso substrato in silicio sia circuiti elettronici sia dispositivi opto-meccanici. La combinazione della microelettronica e dei componenti meccanici rende i MEMS più potenti e versatili rispetto ai sensori tradizionali. I vantaggi sono notevoli e sono dati non solo dalle dimensioni dell'ordine dei micrometri, da cui deriva il nome, ma anche dal miglioramento delle prestazioni e da una maggiore affidabilità. Vantaggi cui si aggiungono il minor costo e il minore assorbimento di corrente rispetto ad altre soluzioni. L'evoluzione dei MEMS è rappresentata dai Nano-Electro-Mechanical Systems che compiono un ulteriore passo in avanti nella miniaturizzazione di questa tipologia di componenti.

L'esigenza di creare dispositivi sempre più piccoli è dovuta soprattutto alle richieste dell'industria dei dispositivi portatili, quali smartphones e wearables, che necessita

sempre più funzioni e capacità computazionale, a parità di peso e dimensioni. I componenti elettronici tradizionali, a montaggio a terminale passante (THT) stanno lentamente scomparendo a vantaggio di quelli SMD (Surface Mounting Device), a montaggio superficiale. Anche questi ultimi però, dalle iniziali dimensioni paragonabili a quelle dei componenti tradizionali, sono ormai minuscoli e caratterizzati da una piedinatura con contatti fitti, che ne rende pressoché impossibile la saldatura senza unirne due o più con la lega saldante, a meno di attrezzature specifiche.

Poiché l'elettronica ha fatto e continua a fare molti progressi nell'integrazione delle funzioni e nella riduzione delle dimensioni dei componenti si è quindi reso necessario rendere accessibili questi dispositivi anche nell'ambito consumer. In questi casi vengono quindi in aiuto le *breakout board*, ovvero schede di prototipazione contenenti uno o più componenti già saldati, le cui connessioni sono portate all'esterno delle schede su connessioni facilmente utilizzabili per passo e terminazione; di solito i collegamenti sono piazzole a passo standard di 2,54 mm, o $1''$, che rendono semplice l'accesso all'interfaccia di comunicazione dei componenti e la lettura, o scrittura, dei dati.

Al fine di questa tesi i movimenti del cubo che interessa rilevare, e sui quali ci si concentrerà nei prossimi paragrafi, sono i seguenti:

1. L'inclinazione e più in generale la rotazione.
2. Lo *shake* o scuotimento.

Pertanto sono necessari due sensori, un accelerometro e un giroscopio, che trovano spazio sull'unità di misura inerziale utilizzata in questo progetto. Sullo stesso modulo si trova inoltre un magnetometro, che nello sviluppo attuale del sistema non è stato utilizzato, ma la sua presenza è fondamentale in applicazioni di manipolazione di oggetti virtuali o di navigazione in spazi 3D in quanto permette di dotare il dispositivo di un sistema di riferimento assoluto. Attraverso questi tre sensori è quindi possibile tenere sotto controllo qualsiasi aspetto relativo al movimento nello spazio compiuto da un dispositivo, sia esso autonomo o manipolato da un utente.

Nel primo paragrafo verrà introdotta la breakout board LSM9DS1 di Adafruit, sulla cui scheda è presente l'omonima IMU prodotta da STMicroelectronics. Su questa breakout dalle dimensioni di appena 33×20 mm e rivolta prevalentemente all'ambito consumer, si trovano un accelerometro e un giroscopio, trattati nei paragrafi successivi, e un magnetometro.

6.5.1 IMU Adafruit LSM9DS1

Il modulo LSM9DS1 di STMicroelectronics consiste in un blocco LGA di dimensioni pari a $3{,}5 \times 3 \times 1{,}0$ mm, sulla cui superficie trovano posto 24 contatti. In tale

IMU sono presenti tre diversi sensori digitali: un accelerometro, un giroscopio ed un magnetometro. Questi sensori forniscono i valori dell'accelerazione lineare, della velocità angolare e dell'intensità del campo magnetico lungo tre assi, e pertanto il sistema è definito da nove variabili indipendenti da cui deriva l'indicazione *9 DOF*, spesso associata a tali moduli. Il numero di gradi di libertà di un sistema meccanico è dato dal numero minimo di coordinate indipendenti necessarie a definire completamente le posizioni di tutte le parti del sistema in qualsiasi istante di tempo [23]. La posizione e l'orientamento nello spazio di un corpo rigido sono definite da tre componenti di traslazione e tre componenti di rotazione, pertanto il corpo dispone di sei gradi di libertà.

Le unità di misura inerziali a sei o a nove gradi di libertà trovano ampio utilizzo nei dispositivi di gioco, nel riconoscimento delle gesture, nelle interfacce interattive e nelle applicazioni di orientamento e navigazione indoor.

Per rendere accessibile nel mondo consumer questo modulo, così come altri componenti con dimensioni analoghe, si rende necessario l'utilizzo delle breakout board. Al fine di questa tesi si è scelta la versione della scheda di prototipazione realizzata da Adafruit, dalle dimensioni di 33×20 mm e mostrata in figura 6.20. Questa breakout board offre accesso sia all'interfaccia I^2C sia a quella SPI, oltre che ai pin di alimentazione e ad alcune funzionalità che possono essere utili per la gestione del programma, come gli interrupt.

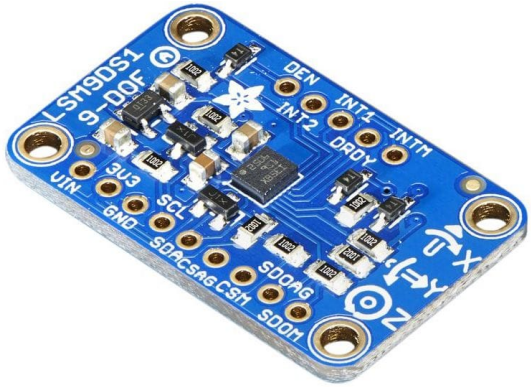


Figura 6.20: Breakout board LSM9DS1. Fonte: Adafruit.

Le caratteristiche principali del modulo LSM9DS1 sono le seguenti:

- accelerometro, giroscopio e magnetometro a 3 assi;
- funzioni di rilevamento della posizione e del movimento;
- interfaccia I^2C (Standard, a 100 kHz, e Fast Mode, a 400 kHz) e SPI;
- programmazione della generazione di interrupt;

- implementazione FIFO per l'output dei dati;
- convertitore analogico-digitale a 16 bit.
- funzionalità di risparmio energetico;
- sensore di temperatura integrato;

Il consumo di corrente del modulo, alimentato a 3,3V, con il magnetometro in modalità *High Resolution* (Output Data Rate = 20 Hz) e l'accelerometro e il giroscopio in modalità normale è di 4,6 mA. Disattivando il giroscopio il consumo ammonta a soli 600 µA. Il consumo di corrente è ad ogni modo diverso a seconda dell'Output Data Rate, o ODR, e di conseguenza della modalità in cui i sensori possono essere configurati: power-down, low-power, normal e high resolution mode.
Il parametro ODR (Output Data Rate), definisce, nei sensori a uscita digitale, la velocità con cui i dati vengono campionati. Strettamente collegato a questo parametro vi è la larghezza di banda, che rappresenta la frequenza più alta di un segnale che può essere campionato senza aliasing, alla velocità definita dall'ODR specificato. La larghezza di banda può mostrare quindi se ad esempio l'accelerometro è in grado di misurare accelerazioni statiche o lente e definisce la frequenza massima f_{max} raggiungibile senza compromettere la precisione della misura. Se il limite inferiore della larghezza di banda, f_{min} non raggiunge 0 Hz l'accelerometro non sarà in grado di misurare accelerazioni statiche come la gravità o vibrazioni lente (< 2 Hz) come quelle che si possono rilevare in ambiente marino. Per il teorema del campionamento di Nyquist, la larghezza di banda è la metà della velocità di campionamento dei dati.

I dati relativi al fondo scala dei tre diversi sensori sono indicati nella tabella 6.2, mentre la sensibilità è riportata nella tabella 6.3.

Altro parametro importante di un'unità di misura inerziale è quello che quantifica l'errore introdotto nella rilevazione quando non vi sono applicate forze, velocità angolari o campi magnetici e che definisce in un certo modo la stabilità dei sensori. Nella tabella 6.4 sono riportati i valori per l'accelerometro, il giroscopio ed il magnetometro presenti nel modulo LSM9DS1.

Il modulo LSM9DS1 dispone di una serie di registri da 8 bit l'uno a cui si può accedere in modalità lettura/scrittura o sola lettura interfacciandosi tramite I^2C o SPI. Ad ogni registro è associato un indirizzo, solitamente in notazione esadecimale, che permette di identificarlo univocamente e di eseguire le operazioni di lettura e/o scrittura.

Questi registri permettono: (1) la configurazione dei sensori e (2) la lettura delle misurazioni effettuate. Attraverso i primi è possibile impostare la modalità di funzionamento, regolare il consumo di corrente, gestire gli interrupt, impostare il fondo scala e l'Output Data Rate, la frequenza di taglio dei filtri passa basso e

Tabella 6.2: Configurazioni del fondo scala dei sensori del LSM9DS1. L'accelerazione di gravità $g = 9{,}81$ m/s^2 viene usata come unità di misura dell'accelerometro. Fonte: STMicroelectronics.

Simbolo	Parametro	Valore	Unità di misura
A_{FS}	Fondo scala dell'accelerometro	± 2 ± 4 ± 8 ± 16	g
G_{FS}	Fondo scala del giroscopio	± 245 ± 500 ± 2000	dps
M_{FS}	Fondo scala del magnetometro	± 400 ± 800 ± 1200 ± 1600	µT

passa alto presenti nel circuito ecc. Per ciascun sensore del modulo, accelerometro, giroscopio e magnetometro, l'output consiste in una terna di valori relativi agli assi x, y e z. Si hanno quindi le misurazioni effettuate da:

- accelerometro: a_x, a_y, a_z;
- giroscopio: G_x, G_y, G_z;
- magnetometro: M_x, M_y, M_z.

Poiché i registri del modulo sono da 8 bit mentre le misure utilizzano 16 bit vengono usati due registri per ciascuna misura relativa ad un singolo asse. In totale quindi il sensore utilizza diciotto registri di sola lettura in cui vengono scritte le misurazioni dei tre sensori, a cui se ne aggiungono altri due che contengono il valore della temperatura corrente. Questi dati vengono poi opportunamente elaborati per attribuire loro un significato all'interno del sistema e rendere tali dati funzionali alle dinamiche di gioco che si approfondiranno nel capitolo 7.

L'accelerometro e il giroscopio hanno due possibili modalità di funzionamento: (1) accelerometro attivo e giroscopio spento o (2) sia accelerometro sia giroscopio attivi, con Output Data Rate uguale. La configurazione può essere modificata tramite il registro CTRL_REG6_XL all'indirizzo $(20)_{16}$ per abilitare la configurazione (1)

Tabella 6.3: Sensibilità dei sensori sul modulo LSM9DS1. Fonte: STMicroelectronics.

Simbolo	Parametro	Configurazione	Valore	Unità di misura
A_{So}	Sensibilità accelerometro	$A_{FS} = \pm 2\,g$	0,06	mg/LSB
		$A_{FS} = \pm 4\,g$	0,12	
		$A_{FS} = \pm 8\,g$	0,24	
		$A_{FS} = \pm 16\,g$	0,73	
G_{So}	Sensibilità giroscopio	$G_{FS} = \pm 245$ dps	8,75	mdps/LSB
		$G_{FS} = \pm 500$ dps	17,50	
		$G_{FS} = \pm 2000$ dps	70	
M_{GN}	Sensibilità magnetometro	$M_{FS} = \pm 400\,\mu$T	14	nT/LSB
		$M_{FS} = \pm 800\,\mu$T	29	
		$M_{FS} = \pm 1200\,\mu$T	43	
		$M_{FS} = \pm 1600\,\mu$T	58	

Tabella 6.4: Errore di offset dei sensori sul modulo LSM9DS1. Fonte: STMicroelectronics.

Simbolo	Parametro	Configurazione	Valore	Unità di misura
A_{Off}	*zero-g offset* dell'accelerometro	$A_{FS} = \pm 8\,g$	± 90	g
G_{Off}	*zero-rate level* del giroscopio	$G_{FS} = \pm 2000$ dps	± 30	dps
M_{Off}	*zero-gauss offset* del magnetometro	$M_{FS} = \pm 400\,\mu$T	± 100	μT

oppure attraverso il registro CTRL_REG1_G all'indirizzo $(10)_{16}$ per la configurazione (2). Il magnetometro ha invece tre possibili modalità di funzionamento: Power-down (default), Continuous-conversion e Single-conversion. Il cambiamento della configurazione è possibile attraverso il registro CTRL_REG3_M all'indirizzo

$(22)_{16}$.

Il modulo LSM9DS1 è collegato al Raspberry Pi secondo lo schema in figura 6.21. Esso adopera l'interfaccia I^2C per comunicare con la scheda Pi Zero W, pertanto utilizza i due pin SDA e SCL, oltre ai pin di alimentazione, 5 V e GND.

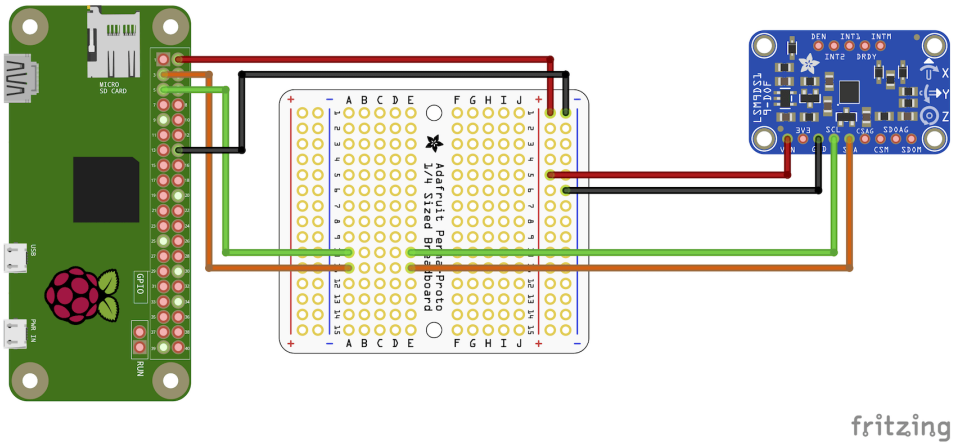


Figura 6.21: Schema di collegamento della IMU LSM9DS1. Realizzato con Fritzing.

6.5.2 Tilt sensing e shake

Tra i modi per interagire con il sistema ed effettuare decisioni all'interno dei giochi proposti sui cubi vi sono la rotazione, l'inclinazione e lo scuotimento dei cubi. Tali modalità di interazione sono rese possibili mediante l'utilizzo di un accelerometro e di un giroscopio. Molteplici applicazioni e giochi sfruttano questi sensori per guidare automobili, pilotare aerei, astronavi e controllare il movimento di personaggi. Alcuni controller di console come il telecomando Wii di Nintendo fanno uso di un accelerometro, che permette di poter usare il controller come mazza da golf, racchetta da tennis o volante virtuale.

Un corpo in stato di quiete, subisce sempre un'accelerazione verso il basso che vale $\simeq 9{,}81\ \mathrm{m/s^2}$. Questo valore di accelerazione, dovuto all'interazione gravitazionale è anche indicato come g. Attraverso la misurazione dell'accelerazione statica dovuta alla forza di gravità, possiamo calcolare l'angolo di inclinazione del sensore con riferimento alla superficie terrestre. Misurando invece l'accelerazione dinamica, si può rilevare la direzione in cui il dispositivo si sta muovendo, oltre all'intensità dello spostamento, e questo è alla base del rilevamento dello scuotimento del dispositivo in una certa direzione.

Generalmente un accelerometro MEMS di tipo capacitivo, quello più comunemente utilizzato in ambito consumer, è la scelta migliore per applicazioni in contesti di

motion sensing, come ad esempio il movimento umano che è relativamente lento con frequenza inferiore a 5 Hz [24]. Al suo interno sono presenti delle microstrutture a pettine collegate ad un punto fisso: quando si applica un'accelerazione a questa struttura, i denti delle microstrutture si avvicinano o allontanano l'uno dall'altro, facendo variare la capacità elettrica tra di essi. Una rappresentazione di tali strutture è presente in figura 6.22. Il sensore converte tali capacità in tensioni, che vengono successivamente convertite attraverso l'ADC presente nel modulo e vengono scritte negli appositi registri.

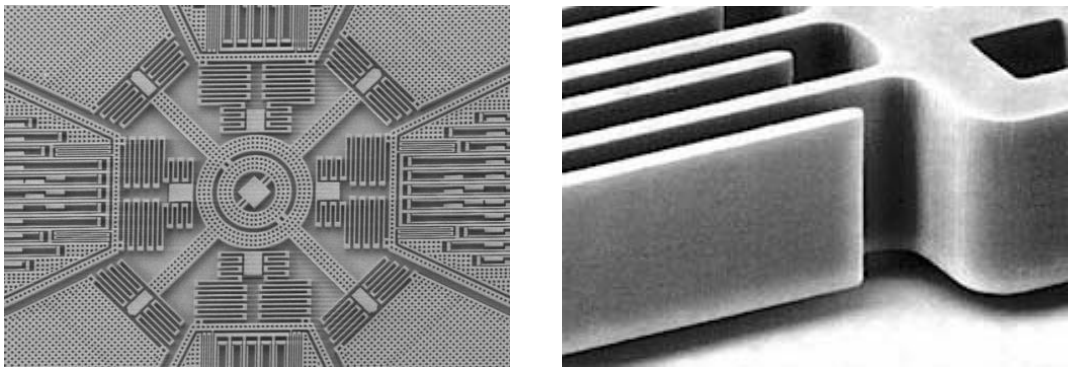


Figura 6.22: Dettaglio delle strutture all'interno di un accelerometro MEMS capacitivo ottenuto tramite un microscopio elettronico a scansione. Fonte: Mikroe.

Gli accelerometri appartenenti a questa tipologia dispongono di un rapporto segnale rumore discreto, di una larghezza di banda limitata e di un range di rilevazione dell'accelerazione tipicamente inferiore ai 100 g. I vantaggi sono dati dal costo molto basso, dall'alta sensibilità e dalla precisione anche ad alte temperature, aspetti che hanno contribuito alla sua diffusione. Offrono inoltre una buona linearità ed un'elevata stabilità del segnale di uscita.

Un corpo può ruotare nello spazio lungo tre assi e facendo riferimento alla dinamica applicata ai velivoli si parla più propriamente di *roll*, *pitch* e *yaw* o rispettivamente rollio, beccheggio e imbardata. La rotazione del velivolo nello spazio è definita dalla terna (ϕ, θ, ψ), figura 6.23. Si ha la seguente convenzione:

- roll (ϕ): rotazione lungo l'asse x;
- pitch (θ): rotazione lungo l'asse y;
- yaw (ψ): rotazione lungo l'asse z.

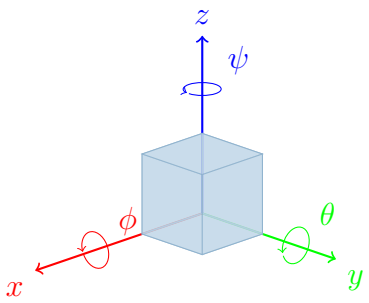


Figura 6.23: Roll, pitch e yaw.

Un accelerometro triassiale è in grado di rilevare l'angolo di roll e l'angolo di pitch ma non l'angolo di yaw, poiché la rotazione intorno a z non provoca una variazione degli angoli che gli assi formano rispetto al piano xy e di conseguenza i valori di accelerazione gravitazionale lungo i singoli assi.
Per tale motivo è necessario usare un giroscopio, che è in grado di fornire l'angolo di rotazione del dispositivo attraverso un'integrazione della velocità angolare $\omega(t)$ rilevata.

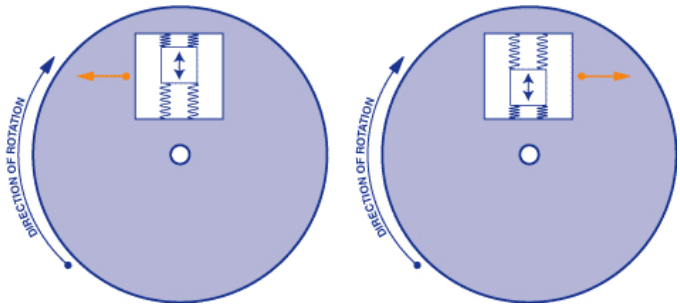


Figura 6.24: Rappresentazione del funzionamento di un giroscopio MEMS.
Fonte: Sparkfun.

L'operazione di calcolo di $\omega(t)$ avviene grazie a delle minuscole masse che si muovono in funzione dei cambiamenti nella velocità angolare. L'operazione di calcolo della velocità angolare in un giroscopio avviene tramite microscopiche masse che si muovono in funzione delle variazioni nella velocità angolare. Queste variazioni vengono quindi opportunamente convertite in tensioni elettriche e amplificate prima di essere rese disponibili negli appositi registri del modulo.

Generalmente ciò che interessa in maggior misura non è tuttavia la velocità angolare $\omega(t)$ ma l'angolo di rotazione $\theta(t)$ del dispositivo. Quindi per ricavare la posizione angolare l'operazione matematica da svolgere consiste in un'integrazione:

$$\theta(t) = \int_0^t \omega(t)dt = \int_0^t \frac{d\theta(t)}{dt}dt \tag{6.2}$$

Trattandosi di sistemi digitali l'integrazione in 6.2 non può essere effettuata ma si deve approssimare tale formula, introducendo quindi un errore nel calcolo, attraverso una sommatoria finita:

$$\theta(t) = \int_0^t \frac{d\theta(t)}{dt}dt \approx \sum_0^t \frac{\Delta\theta(t)}{\Delta t}\Delta t \tag{6.3}$$

Tilt sensing

Un metodo comune per applicazioni di tilt sensing consiste nell'integrazione dell'uscita di un giroscopio, ovvero della velocità angolare rilevata. Questo metodo tuttavia introduce errori che possono rapidamente aumentare e rendere inutilizzabili le rilevazioni del sensore.

Un secondo metodo per determinare l'inclinazione di un dispositivo consiste nell'utilizzo di un accelerometro e sfrutta la rilevazione dell'accelerazione statica, con l'assunzione che l'unica forza agente sul sensore sia la forza di gravità. Scomponendo il vettore dell'accelerazione gravitazionale lungo i tre assi del sensore è possibile stabilire l'angolo di inclinazione del dispositivo.

Calcoli per il tilt sensing

La trattazione che segue si basa sulle note applicative di Analog Devices[25]. Supponendo di avere un accelerometro ad un solo asse sotto le seguenti condizioni, rappresentate in figura 6.25:

1. L'accelerometro è posizionato sul piano xy.
2. L'asse di rilevazione dell'accelerazione del sensore è l'asse x.
3. Il vettore g è orientato lungo l'asse z, con verso negativo.

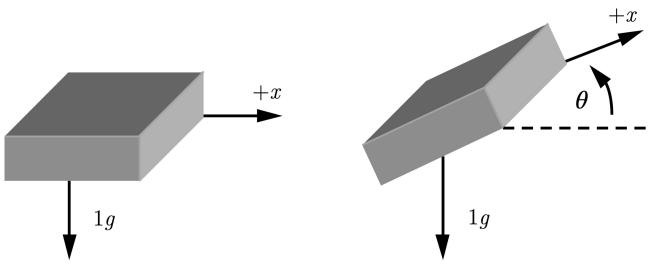


Figura 6.25: Condizioni iniziali dell'accelerometro. Fonte: Analog Devices.

In una condizione ideale, con gravità pari a 1 g, l'accelerazione misurata dal sensore è la seguente:

$$A_{X,OUT} = 1\ g \cdot sin(\theta) \tag{6.4}$$

In questo caso la sensibilità del calcolo dell'inclinazione diminuisce all'aumentare dell'angolo tra il piano xy e l'asse x. Per ottenere l'angolo di inclinazione, in radianti, la formula è la seguente:

$$\theta = \sin^{-1}\left(\frac{A_{X,OUT}}{1\ g}\right) \tag{6.5}$$

Infine, per ottenere l'angolo θ in gradi, si moltiplica il risultato dell'equazione 6.5 per $180/\pi$. L'accelerometro ad un solo asse non può fornire misure a 360 °, poiché l'accelerazione rilevata a t ° coinciderà con l'accelerazione rilevata a $(180 - t)$ °. Di conseguenza per poter rilevare gli angoli di inclinazione lungo un arco di 360 ° è necessario che l'accelerometro disponga di due assi, come illustrato in figura 6.26.

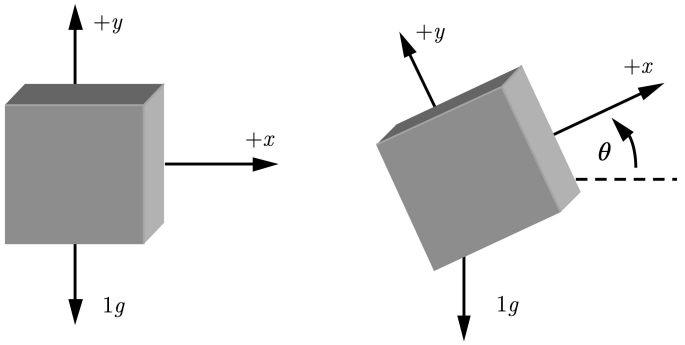


Figura 6.26: Rappresentazione di un accelerometro a due assi. Fonte: Analog Devices.

L'accelerazione $A_{\mathrm{X,OUT}}$ è data dalla 6.4 mentre l'accelerazione $A_{\mathrm{Y,OUT}}$ è:

$$A_{Y,OUT} = 1\ g \cdot cos(\theta) \tag{6.6}$$

Il modo più efficiente per ricavare l'angolo θ, in radianti, è attraverso la 6.7:

$$\theta = tan^{-1}\left(\frac{A_{X,OUT}}{A_{Y,OUT}}\right) = tan^{-1}\left(\frac{1\ g \cdot sin(\theta)}{1\ g \cdot cos(\theta)}\right) \tag{6.7}$$

L'utilizzo di un accelerometro a due assi permette innanzitutto di migliorare la sensibilità del calcolo dell'inclinazione, in quanto le sensibilità dei singoli assi x e y si compensano a vicenda. Inoltre, come anticipato precedentemente, è possibile distinguere l'inclinazione del sensore lungo un arco di 360 °.

Con un accelerometro a tre assi la situazione è leggermente più complessa. Il dispositivo non si muoverà più all'interno di una circonferenza ma all'interno di una sfera. La forza di gravità quindi agirà, a seconda dell'inclinazione, su tre assi. Gli angoli di inclinazione del dispositivo sono mostrati nella figura 6.27, con θ angolo tra il piano xy e l'asse x del sensore, ψ angolo tra il piano xy e l'asse y del sensore e infine ϕ angolo tra l'asse z e l'asse z.

Gli angoli θ, ψ e ϕ vengono ricavati tramite le seguenti equazioni:

$$\theta = tan^{-1}\left(\frac{A_{X,OUT}}{\sqrt{A_{Y,OUT}^2 + A_{Z,OUT}^2}}\right) \tag{6.8}$$

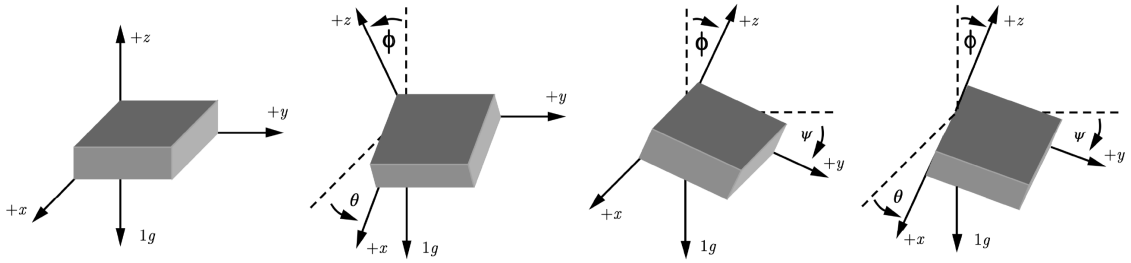


Figura 6.27: Rappresentazione di un accelerometro a tre assi. Fonte: Analog Devices.

$$\psi = tan^{-1}\left(\frac{A_{Y,OUT}}{\sqrt{A^2_{X,OUT} + A^2_{Z,OUT}}}\right) \tag{6.9}$$

$$\phi = tan^{-1}\left(\frac{\sqrt{A^2_{X,OUT} + A^2_{Y,OUT}}}{A_{Z,OUT}}\right) \tag{6.10}$$

Per il sistema CuBee, al fine di rilevare in modo affidabile l'inclinazione dei dispositivi, importante nelle dinamiche di gioco, si è scelto di combinare i valori forniti dal giroscopio con quelli dell'accelerometro, sulle tracce di quanto indicato nel paragrafo corrente.

6.6 Interazione: Modularità

Il concetto di modularità precedentemente introdotto rappresenta la caratteristica principale del sistema CuBee. Con tale termine si fa riferimento al fatto di poter utilizzare un numero arbitrario di cubi e alle varie possibilità di composizione consentite dal sistema.
Al fine di applicare questo concetto è necessario che il singolo cubo sia in grado di:

1. Riconoscere la presenza degli altri cubi connessi sulle proprie facce.
2. Identificare per tali cubi quali siano le rispettive facce coinvolte nella connessione.

Interessa inoltre conoscere l'orientamento assoluto dei singoli cubi in modo da poter ricostruire una mappa virtuale del sistema, utile alla generazione della grafica e alle dinamiche di gioco e bisogna inoltre tenere conto delle seguenti considerazioni, sulla base della figura 6.28:

1. Ogni cubo è dotato di sei facce di cui la faccia E è occupata dal display e pertanto dovrà essere sempre orientata verso l'utente.

2. La faccia F, parallela ad E, non verrà sfruttata, in considerazione di quanto indicato al punto di cui sopra.
3. Le quattro facce rimanenti del cubo, A, B, C, D, dovranno produrre un feedback visivo e solamente su queste facce potranno essere connessi altri cubi e quindi diventare utilizzabili.

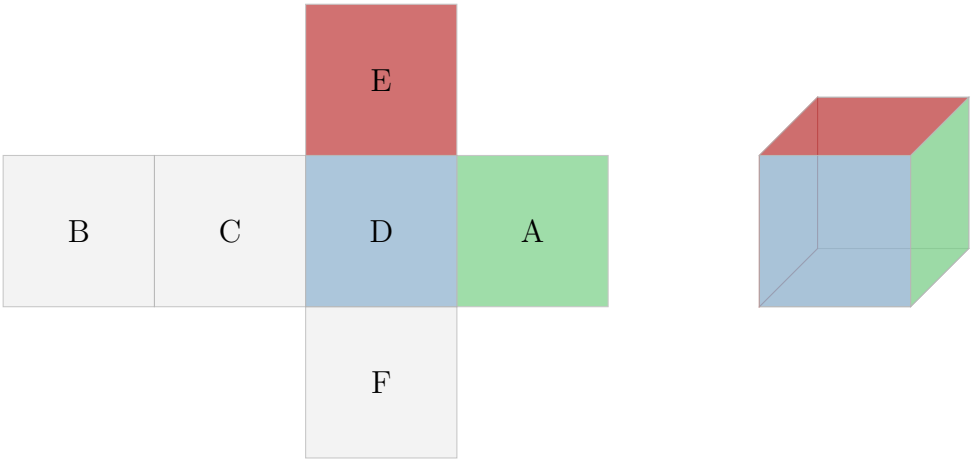


Figura 6.28: Identificazione delle facce dei cubi.

L'implementazione hardware di questa caratteristica fondamentale del sistema CuBee ha posto notevoli difficoltà in fase di progettazione. In particolare, dati due cubi, 1 e 2, con il cubo 1 attaccato sulla faccia B del cubo 2 e quest'ultima collegata sulla faccia A del cubo 1, l'aspetto più complesso da realizzare è stato quello di riconoscere le facce coinvolte nella connessione.

Al fine di mettere in pratica questa funzionalità si sono testate una serie di possibili implementazioni hardware e software. La prima strada intrapresa è stata quella della comunicazione ad infrarossi. Per ciascuna faccia laterale dei cubi si sarebbero dovuti posizionare un ricevitore IR e un trasmettitore, ossia un LED ad infrarossi. Ogni cubo avrebbe quindi utilizzato il LED per inviare un codice univoco della propria faccia e gli altri cubi, in determinate situazioni, avrebbero potuto decodificare tale segnale e riconoscere quindi quale fosse la faccia rivolta verso di esso. Gli svantaggi principali di questa soluzione riguardano in parte la mancanza di stabilità del sistema: infatti si è posto il problema delle false rilevazioni, ovvero in una situazione con un certo numero di cubi sul tavolo è possibile che i cubi non attaccati l'uno all'altro, ma allineati, possano captare il segnale e interpretare tale indicazione come un collegamento fisico dei cubi. Inoltre, dal punto di vista del design, questa soluzione avrebbe compromesso il requisito della semplicità e l'aspetto minimale che il cubo avrebbe dovuto soddisfare, per via dei due fori di circa 4 mm necessari per l'alloggiamento del ricevitore e del trasmettitore sulle facce laterali.

Si è quindi cercata una soluzione diversa sfruttando un componente già presente all'interno del cubo, ovvero l'unità di misura inerziale a nove gradi di libertà. Essendo infatti quest'ultima dotata di un magnetometro, ogni cubo può rilevare il proprio orientamento assoluto e quindi sarebbe stato necessario rilevare unicamente la presenza o meno di un cubo attaccato su una delle proprie facce. Per fare ciò si sarebbe potuto usare un sensore di prossimità che avrebbe indicato la presenza di un ostacolo sulle facce laterali del cubo, ma ciò avrebbe comportato la falsa rilevazione durante la manipolazione dei cubi da parte dell'utente, con l'alto rischio di coprire il sensore e far quindi interpretare al sistema la mano come un cubo. Inoltre sarebbe stato necessario disporre di un foro sulle facce laterali del cubo, equivalente alle dimensioni richieste dalla soluzione precedente.

Mantenendo la possibilità di sfruttare l'orientamento assoluto dei cubi come parte della soluzione, un'altra alternativa possibile per rilevare la presenza di cubi attaccati, senza richiedere fori sul dispositivo, sarebbe stata quella di sfruttare il parametro RSSI (Received Signal Strength Indication) della connessione Bluetooth disponibile sul Raspberry Pi. Questa soluzione è chiaramente quella più ottimale in quanto non prevederebbe ulteriori componenti nel dispositivo, che hanno un impatto sul sistema non solo economico ma anche in termini di requisiti computazionali e complessità aggiunta. Tuttavia questa alternativa è stata scartata in quanto il parametro RSSI relativo alla connessione Bluetooth è influenzato da una serie di variabili ambientali che non avrebbero permesso di determinare una soglia assoluta da utilizzare nel sistema e quindi non sarebbe stato possibile rilevare con precisione il collegamento fisico dei cubi, rendendo instabile il sistema.

Sensore di Hall

In sostituzione dell'indicatore di potenza della connessione Bluetooth si sarebbe potuto utilizzare un sensore di Hall. Questo sensore, estremamente diffuso in molteplici ambiti, come gli elettrodomestici, le automobili, e vari dispositivi portatili, varia la propria tensione in uscita a seconda della polarità o dell'intensità del campo magnetico a cui si trova esposto.

I vantaggi del sensore Hall sono dati dal basso costo, l'affidabilità e le dimensioni ridotte. Il loro tempo di risposta è inoltre molto rapido e se ne possono identificare quattro diverse tipologie [26] che ne rendono possibile l'uso in varie applicazioni:

1. Bipolare (o Latching): il funzionamento di questo tipo di sensore Hall prevede una tensione alta in uscita quando rileva un polo magnetico, mantenendo tale stato fino a quando non rileva un polo magnetico opposto che fa generare al sensore una tensione bassa.

2. Unipolare: il sensore viene attivato unicamente in risposta alla prossimità con un polo magnetico e rimane in tale stato fin quando il campo magnetico persiste. Alla rimozione del magnete pertanto il sensore viene disattivato.
3. Lineare/Analogico: questa tipologia di sensore non prevede un comparatore di soglia con isteresi, che consentirebbe di generare un segnale analogico in uscita variabile tra due soli valori di tensione, per cui il segnale in uscita è strettamente legato all'intensità del campo magnetico a cui il sensore è esposto.
4. Omnipolare: il suo funzionamento è analogo a quello di un interruttore Reed, attivandosi in presenza di un campo magnetico (per entrambe le polarità) e disattivandosi quando il campo magnetico non è più rilevato.

Per quanto riguarda il test sulla modularità effettuato si è utilizzato un sensore Hall unipolare in quanto questa tipologia è stata quella che rispondeva alle necessità di progettazione, oltre ad essere più facilmente reperibile rispetto alla versione omnipolare.
Questa soluzione sarebbe stata ottimale in quanto non avrebbe richiesto fori sulle facce laterali del dispositivo. I sensori ed i magneti inoltre hanno un costo sufficientemente basso quindi non avrebbero avuto un grosso impatto sullo sviluppo complessivo dei dispositivi, e avrebbero costituito un sistema preciso di rilevazione dei cubi collegati.

Tuttavia un'ultima alternativa che si è testata e che infine si è deciso di adottare per lo sviluppo dei prototipi è quella della Near Fied Communication, affrontata nel sottoparagrafo successivo.

6.6.1 Lettore NFC Elechouse PN532 v3

La tecnologia di comunicazione bidirezionale a corto raggio, più comunemente definita NFC, acronimo di Near Field Communication, è una sottocategoria della tecnologia RFID, Radio Frequency Identification, che permette di connettere in modalità wireless due o più dispositivi elettronici posti a breve distanza l'uno dall'altro. Poiché i concetti del NFC derivano da RFID è utile introdurre il principio di funzionamento e le caratteristiche di quest'ultima tecnologia per definire in seguito l'NFC.

RFID

La tecnologia RFID viene utilizzata nella logistica, nella distribuzione ma anche più comunemente nei supermercati come sistema per l'inventario dei prodotti e per la prevenzione contro i furti. Un sistema RFID prevede tre elementi principali: i tag

o transponder, un lettore comprendente un'antenna e infine un sistema di gestione dei dati da e verso il lettore.
I tag RFID possono essere attivi o passivi e la differenza tra le due tipologie è data dalla presenza di alimentazione nel primo caso, mentre per i tag passivi l'energia necessaria per lo scambio dei dati viene generata dal lettore. A livello pratico, ciò comporta che i tag attivi possono coprire una distanza di comunicazione molto più estesa rispetto ai tag passivi, arrivando anche oltre i 100 metri. I vantaggi dei tag passivi d'altro canto sono dati dal costo inferiore e dalle dimensioni minori.

I tag RFID passivi, presenti sotto forma di etichette adesive ad esempio, sono costituiti da un'antenna sotto forma di avvolgimento di un materiale metallico e da un minuscolo circuito integrato che ha la funzione di memorizzare un codice identificativo del tag ed altri dati utili al loro utilizzo. Poiché i tag non hanno una fonte di alimentazione, l'energia necessaria per rispondere ad un'interrogazione del lettore viene fornita dal lettore RFID, il quale emettendo un campo elettromagnetico, tramite un processo di induzione, genera nell'avvolgimento presente nel tag una tensione di alimentazione del microchip.

NFC

I tag NFC condividono lo stesso principio di funzionamento dei tag passivi RFID ma sono pensati per un utilizzo a corto raggio in quanto possono essere rilevati entro una distanza teorica massima di 10 cm, che nella pratica si riduce ad un valore massimo di 3−4 cm. La tecnologia NFC opera ad una frequenza di 13,56 MHz. Lo standard attuale supporta velocità di 106 kbit/s, 212 kbit/s e 424 kbit/s, che risultano sufficienti nella maggior parte dei casi, considerato che le informazioni contenute nel microchip possono arrivare fino ad un massimo di 72 kb. In compenso la tecnologia Bluetooth, che opera a distanze inferiori ai 10 metri, può raggiungere una velocità di 2,1 Mb/s.

A differenza dei tag RFID, i dati presenti nel chip dei tag NFC, in assenza di protezione di scrittura, possono essere riscritti da qualunque lettore. La memoria disponibile all'utente può raggiungere un massimo di 9 kB, a seconda dello standard dei tag.

I tag utilizzati all'interno del progetto sono dotati del chip NTAG215 sviluppato da NXP e conformi ai tag NFC Forum Type 2 e allo standard ISO/IEC14443 Type A [27]. Le caratteristiche del chip sono le seguenti:

- Velocità di trasmissione dei dati: 106 kbit/s.
- Codici identificativi univoci, definiti dal produttore, a 7 byte.
- Protezione della scrittura dati tramite password a 32 bit.

- EEPROM di 540 byte, divisa in 135 pagine da 4 byte ciascuna, di cui 504 byte disponibili per operazioni di scrittura e lettura.

I chip NTAG215 sono inoltre dotati di una funzione di anticollisione che permette di gestire contemporaneamente più di un tag. L'algoritmo di anticollisione seleziona ciascun tag singolarmente e assicura la corretta esecuzione di una transazione con il tag selezionato, senza interferenze da parte di altri tag presenti nel campo di lettura di un dispositivo.

Il dispositivo incaricato di leggere i tag all'interno del cubo si trova su una breakout board prodotta da Elechouse e sulla quale si trova il modulo ricetrasmettitore PN532 di NXP, le cui specifiche sono definite in [28]. Le caratteristiche del modulo sono le seguenti:

- Velocità di trasferimento dati massima: 424 kbit/s.
- Interfacce di comunicazione: SPI, I^2C, High Speed UART (HSU).
- Microcontrollore 80C51 a 8 bit, con ROM di 40 kB e RAM di 1 kB.
- Distanza di lettura massima: 5 cm.
- Modalità a risparmio energetico: Hard-Power Down Mode (2 µA typ, $V_{IN} = 5$ V) e Soft-Power Down Mode (45 µA typ, $V_{IN} = 5$ V).
- Tensione di alimentazione da 2,7 V a 5,5 V.

La breakout board su cui si trova il ricetrasmettitore PN532, visibile in figura 6.29 ha le seguenti dimensioni $43 \times 41 \times 4$ mm. L'antenna incorporata nel PCB, sotto lo strato bianco consente la lettura di tag NFC fino ad una distanza massima di 7 cm. Nella pratica si è riscontrato che nel caso dei tag NTAG215 utilizzati, la distanza di lettura massima, in assenza di ostacoli, raggiunge i 45 mm.

Il modulo PN532 potrebbe agire sia come lettore che come trasmettitore, tuttavia tale funzionamento, per quanto riguarda la breakout board, è possibile unicamente utilizzando l'interfaccia I^2C mentre non è possibile ottenere tale comportamento duale attraverso l'interfaccia SPI, utilizzata nel progetto.

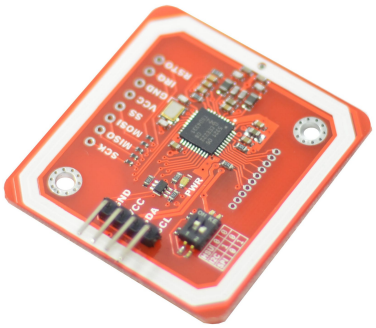


Figura 6.29: Modulo NFC Elechouse PN532

La disposizione dei lettori studiata per ciascun cubo è riportata nel grafico in figura 6.30. Come si può notare, ogni cubo utilizza due lettori NFC inclinati di 45 ° rispetto alla faccia laterale e posizionati negli angoli evidenziati in figura. Per quanto riguarda i tag, ne sono stati utilizzati due tipi diversi, sebbene basati sullo stesso chip NTAG215 di NXP e di uguali dimensioni, 25×25 mm. La

differenza tra i due è data dalla forma diversa dell'avvolgimento e dalla posizione dell'antenna. Il motivo di questa scelta è dato dalla necessità di poter leggere due tag sovrapposti, situazione che si sarebbe verificata in determinate condizioni, a seconda della combinazione dei cubi, per cui in presenza di due tag identici e perfettamente sovrapposti l'uno con l'altro non sarebbe stato possibile leggerli entrambi.

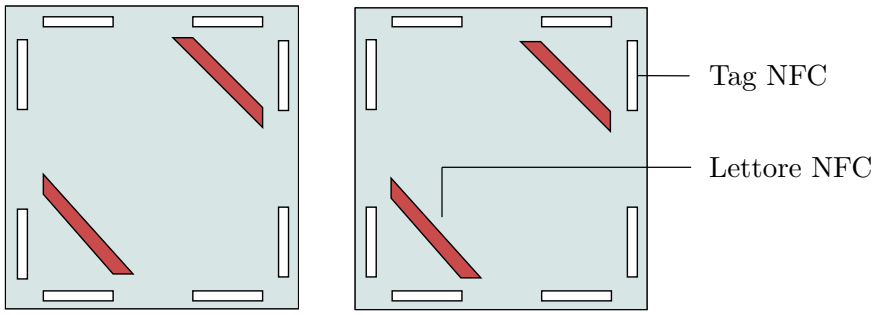


Figura 6.30: Disposizione dei lettori e dei tag NFC all'interno dei cubi.

In ogni cubo sono presenti otto tag NFC, due per ciascuna faccia laterale e la disposizione studiata permette ogni possibile combinazione in quanto vi è sempre un lettore contrapposto ad un tag e quindi indipendentemente da come i cubi sono collegati, il sistema è in grado di ricostruire una mappa dei dispositivi collegati, conoscendo la loro disposizione e organizzando opportunamente la grafica e i contenuti.

Attraverso l'opportuna disposizione utilizzata per i tag e per i lettori NFC, è stato quindi soddisfatto il requisito di modularità del sistema CuBee. Il motivo per cui si è scelto di adottare questa soluzione, rispetto all'utilizzo dei dati forniti dal magnetometro in combinazione con i sensori Hall, è legato alla possibilità di utilizzare NFC anche per lo scambio di dati dinamici senza la necessità di stabilire una connessione a priori. Essendo oggetto del capitolo successivo, la comunicazione tra i cubi verrà quindi ripresa ed elaborata in seguito.

Per ogni cubo sono stati quindi utilizzati due lettori NFC PN532 collegati al Raspberry Pi mediante l'interfaccia SPI. Il collegamento è mostrato nella figura 6.31. I due lettori adoperano la stessa interfaccia SPI del Raspberry Pi e utilizzano uno la linea SPI1 CE1 e l'altro la linea SPI1 CE2.

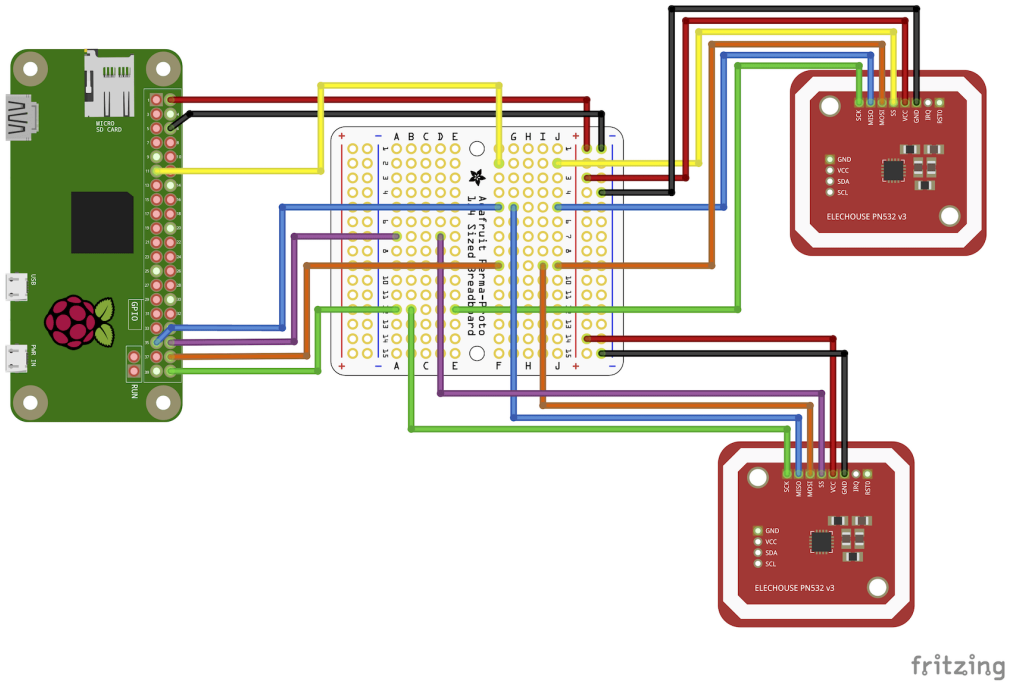


Figura 6.31: Schema di collegamento dei lettori NFC al Raspberry Pi. Realizzato con Fritzing.

6.7 Suono

Il suono ricopre un ruolo importante nei dispositivi progettati in quanto è anche tramite questo aspetto che è possibile rendere l'esperienza immersiva. Per quanto riguarda l'implementazione hardware di questa componente del sistema, le difficoltà principali sono dovute ad una delle caratteristiche del sistema stesso, ovvero il concetto di indipendenza dei cubi. Poiché ogni cubo deve contenere al suo interno i componenti necessari per l'input, l'elaborazione e l'output dei dati, senza richiedere l'ausilio di un ulteriore dispositivo che gestisca parte di questi aspetti, ciò comporta che ogni cubo è dotato di uno speaker e la generazione del suono può quindi essere effettuata da più cubi contemporaneamente. Tale eventualità richiede opportuni accorgimenti in fase di sviluppo del software, soprattutto relativamente al sincronismo tra i dispositivi.

La catena di generazione dell'audio utilizzata in questo progetto prevede un dispositivo di generazione del segnale audio digitale (Raspberry Pi Zero W), un'interfaccia audio PCM che gestisca i flussi dei canali audio in uscita e in ingresso, un convertitore digitale-analogico, un amplificatore e infine uno speaker. La catena audio è del tutto digitale; l'unica conversione in analogico si verifica nell'ultimo passaggio dall'amplificatore MAX98357A allo speaker, che richiede un segnale analogico.

La modulazione a codice di impulsi, PCM, è la forma più diffusa di codifica digitale. Essa si articola in due fasi principali: nella prima fase il segnale viene campionato a intervalli regolari mentre nella seconda fase i valori letti vengono quantizzati, e avviene quindi una rappresentazione discreta del segnale di ingresso rendendone possibile la codifica in digitale.

L'interfaccia audio PCM del Raspberry Pi Zero W, presente sul chip BCM2835, dispone di quattro linee di comunicazione:

1. PCM_CLK o Bit Clock;
2. PCM_FS o Frame Sync Signal;
3. PCM_DIN o Serial Data Input;
4. PCM_DOUT o Serial Data Output.

Le due linee di comunicazione PCM_DIN e PCM_DOUT hanno il compito di trasmettere i flussi audio da e verso l'interfaccia del computer e i dati che vi transitano sono sempre trasmessi a partire dal bit più significativo del frame. Il segnale in transito sulla linea PCM_CLK è invece un segnale di sincronizzazione che rende possibile gestire lo scambio dei bit, mentre il segnale sulla linea PCM_FS è un segnale di sincronizzazione dei frame e viene utilizzato per delimitare i dati seriali in blocchi, di lunghezza variabile e impostabile dall'utente. I frame possono contenere 1 o 2 canali audio sia in uscita che in ingresso al computer. Ogni canale può avere una larghezza compresa tra gli 8 e i 32 bit e può essere posizionato ovunque all'interno del frame, purché i due canali non si sovrappongano.

L'interfaccia audio PCM del chip BCM2835 implementa una struttura dati di tipo FIFO (First In First Out) sia per i canali in ingresso sia per quelli in uscita. Se il frame contiene due canali, questi devono condividere la stesso FIFO e quindi i dati del canale vengono interlacciati.

Nel sottoparagrafo successivo verrà brevemente descritto il protocollo I^2S utilizzato dal Raspberry Pi per interfacciarsi con l'amplificatore MAX98357A descritto in seguito.

6.7.1 I^2S

Il protocollo I^2S, introdotto nel 1986 da Philips Semiconductors, le cui specifiche sono definite in [29], da cui è stata reperita la maggior parte delle informazioni indicate nel presente sottoparagrafo, è un'interfaccia bus seriale che permette di collegare diversi dispositivi e trasferire flussi audio in formato digitale.
Il protocollo prevede almeno tre linee di comunicazione, oltre le quali sono previste delle linee di controllo del trasferimento dati:

1. SD (Serial Data): il trasferimento dei dati, sia in ingresso che in uscita, viene effettuato tramite questa linea.
2. SCK (Continuous Serial Clock): su questa linea transita invece il segnale di sincronizzazione.
3. WS (Word Select) o LRCLK (Left Right Clock): il segnale in transito su questa linea permette di distinguere nel caso di due canali a quale di questi appartiene il frame di dati. Il segnale è a livello logico basso se il frame di dati è relativo al primo canale (Left) oppure è a livello logico alto se è relativo al secondo canale (Right).

I segnali WS e SCK vengono forniti dal master oppure da un dispositivo terzo. Comunemente è il trasmettitore che agisce come master, ma vi è anche la possibilità che il ricevitore si comporti come master oppure tale funzione può essere svolta da un ulteriore dispositivo.

I dati seriali vengono trasmessi utilizzando una rappresentazione in complemento a due, a partire dal MSB. Viene trasmesso per primo il bit più significativo in quanto il trasmettitore e il ricevitore possono avere lunghezze di parola diverse ed ai fini della comunicazione non è necessario né per il trasmettitore conoscere il numero di bit massimo che il ricevitore può gestire, né per il ricevitore conoscere quanti bit sono trasmessi.

Questo protocollo trova uso in buona parte dei microcontrollori, essendo implementato per la trasmissione dei dati audio tra i circuiti integrati.

6.7.2 Amplificatore Adafruit MAX98357A

L'audio generato dal Raspberry Pi è in formato digitale e l'interfaccia PCM del BCM2835 è incaricata della sua trasmissione all'elemento successivo della catena, l'amplificatore, che svolge sia la conversione digitale-analogico sia l'amplificazione del segnale. Il dispositivo utilizzato all'interno di questo progetto è il chip MAX98357A di Maxim Integrated, predisposto su una breakout board realizzata da Adafruit e di dimensioni pari a $19{,}4 \times 17{,}8 \times 3{,}0$ mm. Si tratta di un amplificatore mono di classe D con un'efficienza del 92% ($R_L = 8\Omega$, $P_{OUT} = 1W$) e in grado di generare una potenza di 3.2 W per uno speaker con impedenza pari a 4 Ω.

Le specifiche principali del dispositivo sono le seguenti:

- Potenza in uscita: 3.2 W ($R_L = 4\Omega$, 10% THD, $V_{IN} = 5V$).
- PSRR (Rapporto di reiezione dell'alimentazione): 77 dB typ (a 1 kHz).
- Frequenza di campionamento: 8/ 16/ 48/ 96 kHz.

- 5 livelli di guadagno: 3 dB, 6 dB, 9 dB, 12 dB, 15 dB.
- Distorsione armonica totale + rumore (THD + N) pari allo 0,013% (a 1 kHz).

Sulla breakout board, oltre ai pin di alimentazione Vin e GND, sono presenti i seguenti pin:

1. LRC (Left/Right Clock)
2. BCLK (Bit Clock)
3. DIN (Data In)
4. GAIN: attraverso questo pin è possibile regolare il guadagno che l'amplificatore deve effettuare. Le combinazioni possibili sono le seguenti:
   - 15 dB se vi si connette una resistenza da 100 kΩ tra il pin GAIN e GND;
   - 12 dB se il pin GAIN è connesso direttamente a GND;
   - 9 dB se il pin GAIN non è connesso ad alcun altro pin;
   - 6 dB se il pin GAIN è connesso a Vin;
   - 3 dB se vi si connette una resistenza da 100 kΩ tra il pin GAIN e Vin

Lo schema di collegamento dell'amplificatore, e dello speaker introdotto nel sottoparagrafo successivo, al Raspberry Pi è riportato nella figura 6.32.

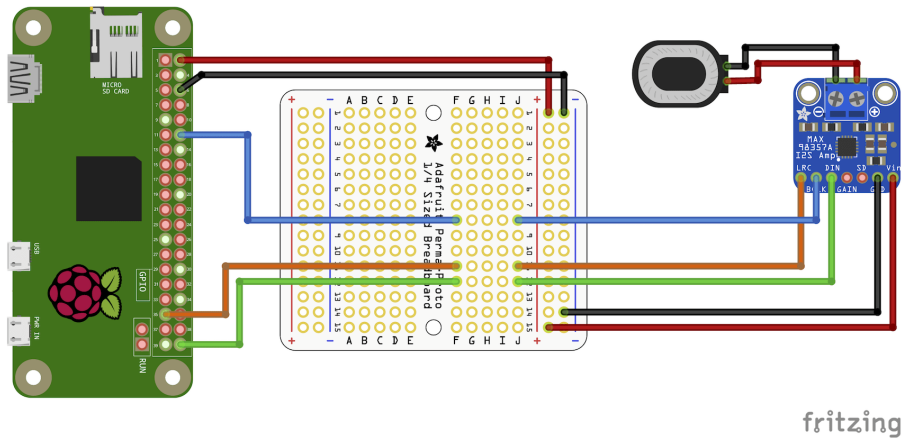


Figura 6.32: Schema di collegamento dei componenti audio al Raspberry Pi. Realizzato con Fritzing.

6.7.3 Speaker CUI CDS-20144-L100

Uno speaker, o altoparlante, è un dispositivo elettroacustico che converte le fluttuazioni di un segnale elettrico in onde acustiche, rendendo possibile la percezione sonora. Dal punto di vista costruttivo, all'interno di un magnete, solitamente al neodimio, viene posto un conduttore in forma di avvolgimento e su questo viene posizionata una membrana mobile detta diaframma. Le fluttuazioni della corrente che attraversano la bobina interagiscono con il magnete, permettendo quindi il movimento del diaframma dell'altoparlante. Tale movimento comporta delle compressioni e dilatazioni dell'aria e attraverso queste oscillazioni viene generato il suono. La membrana è fissata sulla parte superiore dello speaker attraverso un sistema di sospensione elastico che è responsabile del corretto smorzamento delle oscillazioni.

Lo speaker connesso all'amplificatore precedentemente descritto è il modello CDS-20144-L100FF realizzato da CUI e mostrato nella figura 6.33. Si tratta di uno speaker di dimensioni $20 \times 14 \times 3{,}5$ mm con un'impedenza di 4 Ω e che raggiunge un livello di pressione sonora massimo di 95 dB(SPL), valore misurato per un segnale a 1 kHz, con 1 W di potenza assorbita e a 0,1 m di distanza dallo speaker. L'altoparlante tollera una potenza in ingresso fino a 1,2 W.

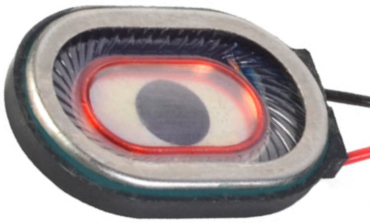


Figura 6.33: Speaker CUI CDS-20144-L100.

Per quanto riguarda la risposta in frequenza, questa è piuttosto ridotta alle basse-medie frequenze, come previsto dalla tipologia di speaker. L'andamento della risposta in frequenza presenta anche variazioni brusche nella banda alta delle frequenze. A differenza delle frequenze alte in cui vi si ritrova in genere poco contenuto informativo, una criticità è data dalle frequenze medio-basse in quanto tra i 100 Hz ed i 300 Hz si trovano le frequenze fondamentali della voce umana e visto l'andamento della risposta in frequenza fino ai 600 Hz ciò potrebbe creare degli inconvenienti dal punto di vista dell'intelligibilità di eventuali istruzioni vocali riprodotte dai dispositivi.

Infine per quanto riguarda la distorsione armonica totale (THD) il valore massimo dichiarato dal produttore, a 1 kHz, è inferiore al 10%. Le criticità maggiori, al pari della risposta in frequenza dello speaker, si verificano per le basse e medie frequenze toccando picchi del 35%.

Il collegamento al Raspberry Pi è mostrato nella figura 6.32 del sottoparagrafo precedente.

6.8 Alimentazione

Il Raspberry Pi Zero W, ovvero il cuore del dispositivo sviluppato, necessita di essere alimentato non solo per svolgere le proprie funzioni base, ma anche per fornire a sua volta l'apporto di corrente necessario ai componenti elettronici finora descritti. La scheda Raspberry Pi richiede una tensione di 5 V e il cavo di alimentazione dovrebbe poter fornire, secondo la Raspberry Foundation, una corrente pari a 2,5 A.

I requisiti legati all'alimentazione del dispositivo progettato dipendono chiaramente dai componenti collegati al dispositivo di elaborazione, dai programmi in esecuzione e dalle funzioni abilitate su tale dispositivo. Per il progetto di questa tesi è necessaria una batteria ricaricabile che soddisfi i requisiti energetici di tutti i componenti presenti all'interno di ciascun cubo, e che permetta di coprire idealmente 8 ore di uso non continuo del dispositivo, garantendo quindi un'autonomia funzionale pari alla tipica giornata di apertura del museo in cui verrà utilizzato.

Per quanto riguarda la scheda Raspberry Pi Zero W, il consumo dichiarato dalla Raspberry Foundation è di 150 mA in assenza di programmi in esecuzione e componenti collegati ai propri pin. Considerando i consumi dei diversi componenti utilizzati e trattati nei capitoli precedenti, nonché quelli della scheda Raspberry Pi, e tenendo in considerazione anche lo spazio disponibile all'interno del cubo così come l'aspetto prettamente economico, un buon compromesso sulla capacità della batteria è stato raggiunto scegliendo una batteria di 4000 mAh. Tale batteria soddisfa le richieste di alimentazione dei componenti elettronici utilizzati e ha permesso di superare le 10 ore di funzionamento del dispositivo in condizioni di test.

La batteria scelta è un tipo di batteria litio-ione-polimero, indicata più comunemente con la sigla LiPo, di dimensioni pari a 70×30×5 mm e con un peso di 95 g. L'uscita della batteria è compresa tra una tensione di 4,2 V (batteria completamente carica) e una tensione di circa 3,7 V. Al suo interno è presente un circuito di protezione il cui compito è quello di evitare che la tensione raggiunga livelli troppo elevati (fenomeno definito *overcharge*) o troppo bassi (*over-use*).

L'uscita della batteria fornisce quindi una tensione pari a 3,7 V, non sufficiente per alimentare il Pi Zero W che richiede 5 V. A tal fine, in mancanza di alternative con pari caratteristiche di ingombro e capacità ma con una tensione maggiore, è necessario utilizzare un convertitore step-up.

6.8.1 Boost converter Adafruit PowerBoost 1000C

Un convertitore step-up, detto anche boost, genera in uscita una tensione maggiore di quella in ingresso. Come conseguenza dell'aumento di tensione V in uscita,

viene ridotta la corrente I, in modo da soddisfare il principio di conservazione della potenza $P = VI$.

Questi convertitori contengono comunemente un diodo, un transistor, un induttore e un condensatore. L'energia viene immagazzinata nell'induttore durante l'attivazione dell'elemento di commutazione (stato ON) e l'induttore si comporta come un generatore di tensione, per cui la sua tensione si somma a quella di ingresso che viene trasferita al condensatore quando la commutazione viene disattivata (stato OFF). Il guadagno di tensione dipende da numerosi fattori, come il duty cycle, il valore dell'induttore, la tensione in ingresso, il periodo di commutazione e la corrente in uscita. Lo schema di funzionamento di un convertitore boost è riportato nella figura 6.34.

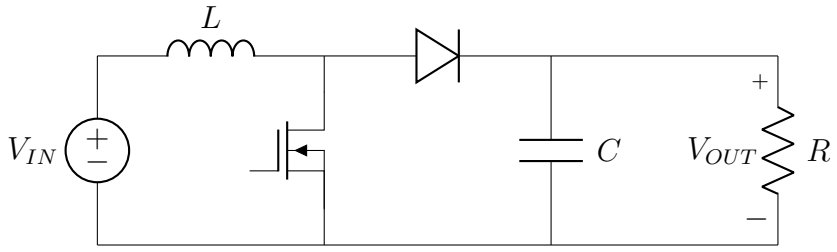


Figura 6.34: Schema di un convertitore boost

Il vantaggio principale di questi convertitori è l'efficienza: alcuni dispositivi possono arrivare fino al 99% di efficienza, ossia disperdono nel processo di conversione solo l'1% della corrente in ingresso[30]. Il PowerBoost 1000 Charger di Adafruit, utilizzato nel progetto, e basato sul convertitore boost TPS61090 di Texas Instruments[31] raggiunge un'efficienza del 96%. Questo convertitore permette di convertire la tensione di qualsiasi batteria LiPo a 3,7 V in una tensione di 5,2 V e può generare una corrente massima pari a 1 A, sufficiente per le esigenze di alimentazione dei componenti presenti nel cubo.

Sulla scheda si trovano tre fonti di alimentazione: l'input microUSB per caricare la batteria (4,75 – 5,25 V), la batteria LiPo (3 – 4,2 V) e l'output (5 – 5,2 V). Uno dei vantaggi del PowerBoost 1000 Charger è il passaggio automatico all'alimentazione USB quando questa è disponibile, invece dell'uso continuo della batteria; permette in questo modo di aumentare il ciclo di vita della batteria, quindi la sua durata. Sulla scheda del convertitore sono presenti inoltre una serie di pin:

- **USB**: è il pin di alimentazione della porta microUSB a 5 V, utilizzato per caricare la batteria e può essere usato per prelevare la corrente dalla porta microUSB quando questa è collegata ad una fonte di alimentazione

- **BAT**: è connesso direttamente al connettore JST della batteria collegata al convertitore e la tensione varia a seconda della batteria, da 3,0 V (batteria quasi esaurita) a 4,2 V (batteria completamente carica).
- **VS**: se dalla porta microUSB arriva una tensione di 5 V da una fonte di alimentazione esterna, su questo pin sarà presente la stessa tensione pari a 5 V. Se il dispositivo non è alimentato tramite USB, il pin avrà invece la stessa tensione della batteria (pin BAT).
- **5V**: è il pin da cui si può prelevare l'output del convertitore, ossia una tensione pari a circa 5,2 V che scende a 5 V all'aumentare della corrente assorbita (oltre i 500 mA).

Vi è inoltre il pin GND che fornisce la massa, comune sia per l'input sia per l'output.

In aggiunta ai pin appena elencati ve ne sono due che hanno delle funzionalità di controllo del convertitore. Il primo di questi è il pin EN (Enable) che permette di attivare o disattivare il convertitore. Esso è collegato a VS e quindi il booster di default è attivo, mentre per disattivarlo è possibile connetterlo a GND e in questo modo l'output del convertitore verrà completamente sconnesso dall'input.
Il secondo pin è chiamato LBO (Low Battery Output) ed è connesso a BAT ma quando il dispositivo rileva un valore inferiore a 3,2 V la sua tensione viene portata a 0 V. Esso permette quindi di conoscere quando la batteria raggiunge un livello critico e prendere di conseguenza degli opportuni accorgimenti quali ad esempio la notifica dello stato della batteria all'utente o lo spegnimento in sicurezza del dispositivo. Il collegamento al Raspberry Pi è mostrato nella figura 6.35.

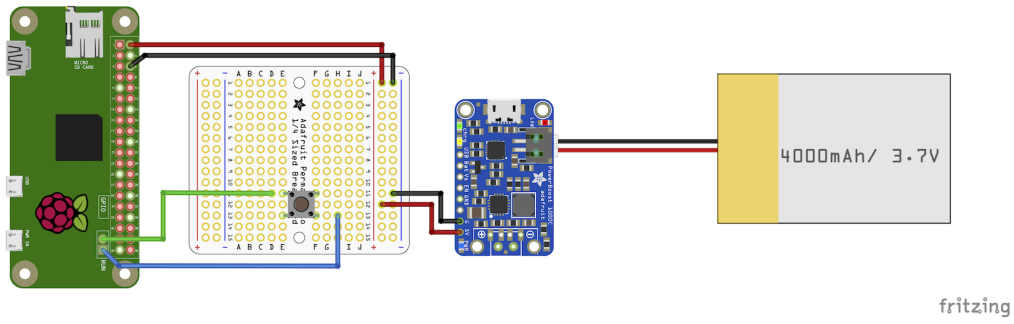


Figura 6.35: Schema di collegamento del PowerBoost 1000C al Raspberry Pi. Realizzato con Fritzing.

Capitolo 7

Progettazione software

— Stelian Spulber

In questo capitolo verranno trattati alcuni argomenti relativi all'ambito software del sistema CuBee, ossia l'insieme delle operazioni svolte e dei programmi scritti per essere eseguiti dal Raspberry Pi al fine di generare l'interfaccia dei dispositivi sviluppati, in base ai segnali ricevuti dai componenti hardware e in base alle operazioni necessarie per le diverse modalità di gioco studiate.

Innanzitutto verranno introdotti alcuni dei possibili protocolli di comunicazione utilizzabili nell'ambito IoT, tra cui il protocollo utilizzato per il sistema in oggetto, e in seguito verrà affrontato il principio di funzionamento e le scelte effettuate per la programmazione del dispositivo, nonché per la realizzazione dei giochi previsti nell'installazione museale.

I videogiochi e il codice incaricato della gestione dei dispositivi utilizzano il linguaggio Python[1]. Questo linguaggio di programmazione ad alto livello, reso pubblico nel 1991 da Guido van Rossum e gestito dalla Python Software Foundation, supporta diversi paradigmi di programmazione, incluso quello Object-Oriented, ed è semplice da apprendere ed utilizzare in quanto nato per essere immediatamente intuibile grazie ad una sintassi pulita e ai blocchi logici costruiti utilizzando la stessa indentazione. Si tratta di un linguaggio pseudocompilato, nel quale l'interprete analizza il codice sorgente e lo esegue, qualora non vi siano errori sintattici; inoltre è un linguaggio portabile, che può essere interpretato ed eseguito su molteplici piattaforme, tra cui Windows, Mac OS, Linux ecc.
Secondo il TOBE Index, Python risulta essere il terzo linguaggio di programmazione più popolare nel 2019, mentre il PYPL Index ne conferisce il titolo di linguaggio di programmazione più ricercato su Google a febbraio 2020 [2].
Si è scelto di utilizzare in questa prima fase la versione Python 2.7.16, sebbene

sconsigliata e a breve non più ufficialmente supportata, per motivi di compatibilità con alcune delle librerie utilizzate.

Nei paragrafi successivi, i concetti teorici brevemente trattati saranno accompagnati dalla relativa implementazione nel sistema CuBee, nonché nei videogiochi sviluppati.

7.1 Comunicazione tra i cubi

Il primo aspetto che si è scelto di affrontare riguarda la comunicazione tra i singoli dispositivi di gioco. Si è infatti reso necessario mettere in atto uno scambio di dati costante tra i cubi, i quali devono essere a conoscenza dei diversi stati in cui si può trovare il sistema e le dinamiche di gioco devono essere condivise in tempo reale.

I dati che devono essere scambiati all'interno del sistema CuBee possono essere definiti *event-driven*, ossia la trasmissione di un nuovo dato avviene solo ed esclusivamente in seguito al verificarsi di un determinato evento o condizione, come nel caso di un cubo che viene affiancato ad un altro. Tuttavia, nel caso di alcuni giochi si rende necessario far conoscere lo stato del sistema e la disposizione dei cubi ad ogni dispositivo attivo, pertanto una comunicazione event-driven non è più sufficiente, ma occorre trasmettere i dati ad intervalli regolari.
I dati che vengono generati all'interno del sistema possono essere di due tipologie: dati relativi alle dinamiche del gioco, ad esempio il punteggio o eventi analoghi che devono essere inviati a tutti i dispositivi, e dati relativi allo stato del sistema, ad esempio l'inizio dell'interazione, l'attivazione e la disattivazione dei cubi o l'esito del gioco, quest'ultimo fondamentale per modificare l'evoluzione della storia del cortometraggio di animazione.

Dal punto di vista dei protocolli di comunicazione, la scheda Raspberry Pi dispone sia di connettività Wi-Fi che Bluetooth (e Bluetooth Low Energy), pertanto è possibile implementare un'ampia gamma di protocolli, quali TCP, UDP, MQTT, Bluetooth ecc.

Bluetooth Low Energy

Il Bluetooth è una tecnologia wireless di trasmissione dati il cui standard viene periodicamente aggiornato dal Bluetooth Special Interest Group (SIG). Si tratta di una tecnologia di comunicazione a corto raggio, operante nella banda dei 2,4 GHz e caratterizzata da un ridotto consumo energetico. Vi si possono distinguere due diverse versioni: Basic Rate/ Enhanced Data Rate (BR/ EDR) e Low Energy (LE)[3]. Il protocollo Bluetooth Low Energy è stato introdotto nel 2010, con la

versione 4.0 del documento Bluetooth Core Specification [3]. Bluetooth LE è caratterizzato da un consumo energetico notevolmente inferiore rispetto al Bluetooth classico (BR/ EDR).
Una delle componenti principali dello stack Bluetooth LE è il Generic Access Profile (GAP) che definisce i requisiti fondamentali di un dispositivo Bluetooth e descrive il comportamento e i metodi necessari ad instaurare una connessione, la sicurezza e l'autenticazione e il servizio di *discovery*. Il GAP definisce quattro ruoli possibili:

- Broadcaster: prevede unicamente l'advertising dei dati e non prevede la possibilità di instaurare una connessione con un altro dispositivo.
- Observer: ruolo complementare al precedente, Broadcaster, e quindi anche questo ruolo non permette di stabilire una connessione con un altro dispositivo.
- Peripheral: ruolo assunto da dispositivi che supportano una singola connessione Bluetooth, come ad esempio uno speaker, un fitness tracker ecc. Un dispositivo che agisce come periferica può avere due stati: advertising e connected.
- Central: permette di instaurare più connessioni contemporanee. Un dispositivo che agisce come centrale può avere due stati: *scanning* e *connected*.

Il profilo GATT (Generic Attribute Profile) definisce la struttura dei dati e le modalità con cui questi dati vengono scambiati tra due dispositivi Bluetooth. Il GATT definisce i seguenti ruoli:

- Client: invia le richieste al server e riceve le risposte da quest'ultimo. Per poter leggere e/o scrivere gli attributi presenti sul server, il client deve effettuare l'operazione *service discovery*, scoprendo quindi i servizi disponibili.
- Server: riceve le richieste da un client e fornisce le risposte all'interrogazione.

I ruoli GATT sono completamente indipendenti dai ruoli GAP e un dispositivo con ruolo Central (GAP) o Peripheral (GAP) può agire sia come client sia come server (GATT) e può eventualmente assumere il ruolo di client e server contemporaneamente.
Vengono definiti inoltre i seguenti concetti:

- Service: contenitore che raggruppa una serie di caratteristiche correlate. Dispone di un identificativo univoco, UUID, di lunghezza pari a 128 bit.
- Characteristic: si compone di una parte dichiarativa, in cui sono presenti una serie di metadati, incluse alcune proprietà e configurazioni della caratteristica come ad esempio la possibilità di modificarne i dati o di poterla utilizzare all'interno di un *advertising packet*, e una parte che ne contiene i dati. Le caratteristiche, così come i servizi, sono identificate da un UUID di lunghezza pari a 128 bit.

- Descriptor: ulteriore campo che permette di estendere la cosiddetta *characteristic declaration*, ossia i metadati della caratteristica.

Uno degli aspetti più interessanti del protocollo Bluetooth LE riguarda l'Advertising. Questa operazione consiste nel trasmettere in broadcast e ad intervalli regolari un pacchetto di dati relativi al dispositivo in questione. Tali pacchetti possono quindi essere letti da qualunque dispositivo che agisce come Observer o Central.
L'adozione del protocollo Bluetooth LE permetterebbe quindi di ridurre il consumo energetico dei dispositivi rispetto all'utilizzo di una connessione Wi-Fi, oltre a rendere non più necessaria la creazione di una rete per lo scambio di dati tra i singoli cubi e tra i cubi e il computer che gestisce l'interazione con il cortometraggio proiettato.

Per quanto riguarda il Raspberry Pi, e più in generale Linux, il supporto al protocollo Bluetooth è gestito dal kernel BlueZ, parte del kernel ufficiale di Linux a partire dalla versione 2.4.6.

MQTT

Un ulteriore protocollo di comunicazione utilizzato in ambito IoT è MQTT, MQ Telemetry Transport, il cui sviluppo è attualmente gestito da OASIS. Si tratta di un protocollo snello e particolarmente semplice, sviluppato per dispositivi con caratteristiche computazionali ridotte e larghezza di banda bassa.
MQTT prevede tre ruoli: *publisher*, *subscriber* e *broker*. Il broker è rappresentato dall'applicativo incaricato di ricevere i messaggi da una serie di dispositivi, definiti publisher, e di consegnare tali messaggi ai dispositivi chiamati subscriber. I client possono iscriversi a una serie di argomenti, *topics*, in modo da essere notificati dal broker quando vi sono nuovi dati.

MQTT fornisce tre livelli di Quality of Service che possono essere usati in modo da rendere affidabile il trasferimento dei dati tra il broker ed i client:

- 0: il broker/client consegna il messaggio una sola volta, senza richiedere alcuna conferma.
- 1: il broker/client consegna il messaggio almeno una volta, richiedendo conferma di ricezione al destinatario.
- 2: il broker/client consegna il messaggio una sola volta, utilizzando un processo di handshake in 4 passi.

Il livello di QoS è definito per ciascun messaggio e client, e chiaramente più è alto il livello impostato maggiore sarà l'affidabilità della trasmissione ma sarà maggiore anche la latenza e la richiesta di larghezza di banda.

OSC

Il protocollo Open Sound Control, OSC, è stato messo a punto dal Center for New Music and Audio Technology, CNMAT, della Berkeley University nel 1997, ed è un protocollo open-source nato per lo scambio di dati in tempo reale soprattutto in ambito musicale. Si tratta di un protocollo indipendente dal layer di trasporto e dalla piattaforma, e può essere usato con UDP/IP, TCP/IP, connessioni seriali o all'interno di applicazioni. I vantaggi di questo protocollo sono dati dalla facilità di implementazione e dalla flessibilità offerta.

OSC è stato progettato con un'architettura del tipo client/server, dove il client è il dispositivo che invia una serie di dati mentre il server è il dispositivo che riceve tali dati. Le tipologie principali di dati scambiabili sono: numeri interi (int32), numeri a virgola mobile (float32), stringhe e timetag; ciascuna tipologia utilizza multipli di 4 byte. I dati vengono scambiati sotto forma di pacchetti, OSC Packet, i quali sono composti da un blocco contiguo di dati, rappresentati in formato binario, e dalla dimensione in byte di tale blocco. La dimensione di un OSC Packet, in byte, è sempre un multiplo di 4.
Il contenuto principale di tali pacchetti prende il nome OSC Message ed è composto dai seguenti argomenti:

- OSC Address Pattern: stringa che inizia con il carattere `</>` (forward slash) e costituisce un identificativo del messaggio, come ad esempio `/start`.
- OSC Type Tag: stringa che inizia con il simbolo `<,>` (virgola), seguito da una sequenza di caratteri che rappresentano la tipologia degli argomenti del messaggio. I tipi standard previsti dalle specifiche 1.0 del protocollo OSC [4] sono: `i`, `f`, `s` e `t`, che rappresentano rispettivamente i tipi intero, a virgola mobile, stringa e timetag.
- OSC Arguments: sequenza degli argomenti del messaggio; in un messaggio è facoltativa l'indicazione degli argomenti.

In aggiunta all'OSC Message è possibile scambiare anche OSC Bundles, ossia pacchetti composti da uno o più elementi, che possono essere sia messaggi sia OSC Bundles. Questi contenitori sono identificati dalla stringa iniziale `#bundle` di dimensione 8 byte, e sono inoltre composti da un timetag di dimensione 8 byte, e di una sequenza di OSC Messages e/o OSC Bundles. Non è quindi presente un indirizzo identificativo del bundle, bensì tale indirizzo sarà compreso negli eventuali messaggi contenuti all'interno del bundle.

La libreria utilizzata per il sistema CuBee è pyOSC, che implementa le specifiche del protocollo OSC ed utilizza il protocollo UDP/IP per l'invio e la ricezione di pacchetti conformi alle specifiche OSC. Sebbene sia supportato anche il protocollo TCP/IP, pressoché tutte le implementazioni di OSC utilizzano UDP/IP in quanto

questo protocollo è nato per l'utilizzo in ambito musicale, con il vincolo del *real-time*, in cui interessa avere una latenza più bassa possibile e un flusso di dati ordinato e continuo.

Alla base del funzionamento della libreria si trova il concetto di *socket*, ossia quell'interfaccia software che permette lo scambio di dati tra host remoti, attraverso una rete, o tra processi che avvengono localmente, Inter-Process Communication. In Python l'interfaccia socket è accessibile mediante l'omonimo modulo, `socket`.
Le due principali famiglie, o domini, di socket sono le seguenti:

- Internet domain: riguarda il trasferimento di dati tra processi in esecuzione su host remoti.
- UNIX domain: riguarda lo scambio dei dati tra processi che avvengono sullo stesso host.

I socket inoltre possono essere suddivisi in due principali tipologie:

- Stream: permettono un flusso di dati bidirezionale e affidabile, stabilendo una connessione tra gli host basata sul protocollo TCP/IP.
- Datagram: non garantisce l'affidabilità offerta dai Stream socket in quanto si basano sul protocollo UDP/IP e quindi non viene instaurata una connessione tra gli host ma i dati vengono trasmessi sotto forma di datagrammi.

Con la libreria pyOSC, vengono utilizzati i Datagram socket e l'Internet domain. La creazione di un socket in Python avviene come indicato di seguito:

```
import socket

# crea un socket nell'Internet domain, utilizzando UDP/IP
d = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
# crea un socket nell'Internet domain, utilizzano TCP/IP
s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
```

PyOSC definisce le seguenti classi principali:

- `OSCMessage(address,args)`: definisce l'oggetto di tipo OSC Message che fornisce i metodi necessari alla costruzione di un messaggio conforme alle specifiche del protocollo OSC. Un oggetto `OSCMessage` può essere configurato con l'indirizzo `address` e gli argomenti `args` in fase di dichiarazione oppure in un momento successivo, mediante i metodi `setAddress(address)` e `append(args)`.
- `OSCBundle()`: definisce l'oggetto di tipo OSC Bundle che fornisce i metodi necessari alla costruzione di un bundle.

- `OSCClient()`: gestisce l'invio dei pacchetti OSC, fornendo i metodi per connettersi a un determinato server identificato da `addressC = (hostname, port)`, per inviare un messaggio, `send(msg,timeout)`, ad un server già connesso, con il metodo `connect(addressC)`, o per inviare messaggi a un server al quale non si è connessi, `sendto(msg, address, timeout)`.
- `OSCServer(address)`: è l'oggetto che rimane in ascolto all'indirizzo `address = (hostname,port)` e gestisce la ricezione dei pacchetti OSC, effettuando l'indirizzamento dei messaggi ricevuti.

Vi sono inoltre la classe `OSCMultiClient`, un particolare tipo di `OSCClient` che sfrutta il concetto di Multicast, permettendo di inviare messaggi a più OSCClients (definiti `OSCTargets`), la classe `OSCStreamingServer(address)`, che consente di definire un server connection-oriented (TCP/IP), e l'analoga classe `OSCStreamingClient`.

Interessa tuttavia valutare anche le prestazioni del protocollo MQTT, comparandole con quanto attualmente ottenuto dall'utilizzo di Open Sound Control. In previsione di un possibile utilizzo dei cubi in modalità *standalone*, ovvero senza la necessità di un computer dedicato, è opportuno inoltre valutare nuove modalità di comunicazione tra i cubi che non richiedano, tra l'altro, la connessione ad una rete.

Esempio: *Unlock the cube*

I cubi istanziano un `OSCServer` che ascolta i messaggi `/start` provenienti dal computer e che attivano il gioco indicato dall'argomento del messaggio. Complessivamente, i giochi previsti dal sistema CuBee sono 4, pertanto nel caso del primo gioco, *Unlock the cube*, i cubi riceveranno il messaggio `/start 1`. La porta sulla quale i server restano in ascolto è la 10.001 per il cubo Main (1) e la 10.000 per i cubi secondari (2-5).
Allo stesso modo, per il cubo Main, viene istanziato un `OSCClient` che invia in broadcast sulla porta 10.000 il messaggio `/stop` quando tutti i contenuti dei cubi sono stati sbloccati e quindi il programma in esecuzione sul computer riprende la riproduzione del cortometraggio proiettato e i cubi vengono messi in standby, in attesa di ricevere un nuovo messaggio `/start`.

Gli altri messaggi che vengono scambiati riguardano l'evoluzione del gioco; nello specifico le modalità di gioco prevedono la combinazione di due soli cubi alla volta, per cui un ulteriore messaggio che deve essere inviato dai cubi secondari affiancati uno alla volta al Main riguarda la fine del contenuto mostrato sui display: viene quindi inviato il messaggio OSC `/end`, contenente come argomento il numero identificativo del cubo in oggetto, sebbene questo possa essere anche ricavato dall'indirizzo IP ricevuto con il messaggio. Il cubo Main tiene traccia dei cubi attualmente attivi e non ancora utilizzati, aggiornando una lista, `remCubes`,

ad ogni messaggio `/end` ricevuto. Quando tutti e quattro i cubi secondari sono stati utilizzati e il contenuto previsto è stato visionato, la lista non contiene più alcun elemento e quindi viene segnalato l'evento al computer e ai cubi del sistema, trasmettendo in broadcast il messaggio `/stop`. I cubi vengono messi in standby e riprende la proiezione del cortometraggio.
La dichiarazione e l'utilizzo dell'`OSCServer`, relativamente al cubo Main, sono riportati di seguito:

```
CUBE_ID = 1
Running = False

RX_ADDRESS = ('192.168.0.255', 10001)
TX_ADDRESS = ('192.168.0.255', 10000)

s = OSC.OSCServer(RX_ADDRESS)  # creazione del server OSC
s.addMsgHandler("/start", startHandler)  # se il messaggio contiene
↪  l'indirizzo /start, esegue la funzione startHandler
s.addMsgHandler("/end", endHandler)
serv = threading.Thread(target = s.serve_forever)  # crea un thread
↪  per il server
serv.start()  # avvia il server e rimane in ascolto
```

La funzione `startHandler()` è così definita:

```
def startHandler(addr, tags, data, source):
  global Running
  Running = True  # variabile utilizzata per l'esecuzione del loop
  ↪  principale

  startInit()  # inizializza i lettori NFC
  r = threading.Thread(target = readTags)  # crea un thread per la
  ↪  lettura dei tag NFC
  r.start()  # lancia il processo di lettura dei tag NFC
```

Il messaggio `/stop` inviato al pc e ai cubi secondari viene invece generato nella funzione `stopGame()`, invocata dalla funzione `endHandler()` quando si verifica la condizione `len(remCubes) == 0`, ed è così definita:

```
def stopGame():
  global remCubes
  c = OSC.OSCClient()  # creazione del client OSC
  c.connect(TX_ADDRESS)
```

```
    stopMsg = OSC.OSCMessage("/stop")  # creazione del messaggio OSC
    c.send(stopMsg)  # invio del messaggio

    gameDisplay.fill(BLACK)
    pygame.display.flip()

    Running = False
    remCubes = [2,3,4,5]
```

I cubi secondari ricevono il messaggio `/start` in modo analogo al cubo Main, chiaramente utilizzando la porta 10.000, mentre l'invio del messaggio `/end` viene effettuato dalla funzione `contentEnded()`, invocata alla fine del contenuto riprodotto sul display del cubo:

```
TX_ADDRESS = ('192.168.0.109', 10001)  # 192.168.0.109 è l'indirizzo
 ↪  IP del cubo Main

def contentEnded():
    c = OSC.OSCClient()
    c.connect(TX_ADDRESS)
    stopMsg = OSC.OSCMessage('/end')
    stopMsg.append(CUBE_ID)
    c.send(stopMsg)

    Running = False

standby()
```

In seguito all'invio del messaggio `/end` viene invocata la funzione `standby()` che inizializza il display ed interrompe temporaneamente la scansione dei tag NFC, in modo da preservare la durata della batteria.

Per quanto riguarda la logica del programma eseguito sul computer, che controlla l'interazione dei cubi con il cortometraggio, questa verrà approfondita nel capitolo 8, in cui verranno descritti anche i diversi videogiochi, dal punto di vista dei contenuti e del loro significato all'interno del percorso dell'installazione.

7.2 Implementazione della modularità

La modularità del sistema è stata implementata mediante l'uso della tecnologia NFC, disponendo ciascun cubo di due lettori PN532, opportunamente posizionati

in modo da poter leggere due facce del cubo ciascuno.

I tag utilizzati sono stati innanzitutto memorizzati in un dizionario, `dict`, ossia una struttura dati in cui ogni elemento è definito da una coppia `chiave, valore`. Evidentemente la chiave deve essere univoca, in quanto è attraverso la chiave che vengono recuperati i valori degli elementi memorizzati all'interno del dizionario.
La struttura di tale dizionario, denominato `allTags`, è così definita: `UUID, [cubeId, tagId, faceId]`, con chiave la stringa `UUID` identificativa di ciascun tag, così come definito dal produttore, e come valore un array composto da:

- `cubeId`, numero identificativo del cubo da 1 a 5;
- `tagId`, valore univoco assegnato a ciascun tag;
- `faceId`, valore che indica la faccia laterale a cui è associato il tag e alla nomenclatura introdotta precedentemente, [A, B, C, D], corrispondono i valori [1, 2, 3, 4].

Ogni cubo dispone di otto tag, due per ciascuna faccia e numerati in ordine crescente a partire dalla faccia A; il `tagId` è quindi composto da due cifre di cui la prima è identificativa del cubo, compresa tra 1 e 5, e la seconda rappresenta il numero progressivo del tag, a partire da 1.
Il dizionario `allTags` è quindi così definito:

```
allTags = {
  '040a97f26f5e81': [1,11,1], # primo tag della faccia A
  '040827e2a86481': [1,12,1], # secondo tag della faccia A
  '042198f26f5e81': [1,13,2], # primo tag della faccia B
  '04d512e2a86480': [1,14,2], # secondo tag della faccia B
  '041598f26f5e81': [1,15,3],
  '04433ce2a86481': [1,16,3],
  '040f98f26f5e81': [1,17,4],
  '04101de2a86481': [1,18,4],
  ...
}
```

Ogni cubo utilizza quattro interi per stabilire quali cubi siano affiancati sulle proprie facce: `RIGHT, UP, LEFT` e `DOWN`. Pertanto si è scelto di far assumere a tali valori l'identificativo del rispettivo cubo presente sulla faccia cui la variabile fa riferimento. Ad esempio se tali variabili assumono i valori `[2, 0, 3, 0]` per il cubo Main, esse indicano che sulla faccia A è presente il cubo 2 e sulla faccia C è affiancato il terzo cubo. In questo modo è possibile ricostruire lo stato del sistema nei giochi che sfruttano pienamente le potenzialità del sistema. Ad esempio, nel caso del gioco conclusivo dell'installazione, *End the Strike*, ogni cubo riconosce su, e con, quali facce i cubi sono affiancati, quindi il personaggio rappresentato viene fatto muovere

di conseguenza, iniziando un'azione sul display di un cubo e proseguendo la stessa azione sul display degli altri cubi collegati.

Per la gestione dei lettori NFC basati sul chip PN532 ci si è basati sulla libreria Adafruit_PN532[5]. Tale libreria mette a disposizione la classe `PN532()`, attraverso la quale, con gli opportuni argomenti indicati in seguito, viene istanziato l'oggetto che permette di leggere e scrivere i tag NFC tramite un lettore compatibile collegato alla scheda Raspberry Pi mediante l'interfaccia SPI.
Affinché i lettori possano leggere e scrivere correttamente i tag NFC, essi devono essere opportunamente inizializzati, come descritto di seguito:

```
import Adafruit_PN532 as PN532  # importa il modulo Adafruit_PN532

CS1 = 17
CS2 = 16
MOSI = 20
MISO = 19
SCLK = 21

def startInit()
# instanzia i due oggetti PN532, passando come argomenti i pin usati
 ↪  per la connessione SPI
upperReader = PN532.PN532(cs=CS2, sclk=SCLK, mosi=MOSI, miso=MISO)
  lowerReader = PN532.PN532(cs=CS1, sclk=SCLK, mosi=MOSI, miso=MISO)

  # chiama il metodo begin() che pone a livello logico basso il pin
   ↪  CS per 1 s
  upperReader.begin()
  lowerReader.begin()

  # configura i lettori per poter leggere i tag NFC
  upperReader.SAM_configuration()
  lowerReader.SAM_configuration()
```

La scansione dei tag avviene invocando il metodo `read_passive_target()` dell'istanza dell'oggetto PN532, la quale restituisce un bytearray contenente l'UUID del tag letto, pari a 7 byte. Per poter confrontare i valori degli UUID è necessario convertire in formato esadecimale la rappresentazione binaria restituita dal metodo `read_passive_target()` e ciò viene eseguito tramite la funzione `hexlify()` del modulo `binascii` precedentemente importato.
La funzione `readTags()` gestisce la lettura e il confronto del tag corrente con quelli presenti nel dizionario `allTags`, al fine di identificare quale faccia di quale cubo è connessa al cubo attuale e, tralasciando alcune righe di codice, è così definita:

```
def readTags():
  global upperReader, lowerReader, RIGHT, UP, LEFT, DOWN

  while Running:
    upperTagRead = upperReader.read_passive_target()
    lowerTagRead = lowerReader.read_passive_target()

    if upperTagRead is not None:
      upperTagHex = binascii.hexlify(upperTagRead)
      check = allTags[upperTagHex][1] % 2
      cubeId = allTags[upperTagHex][0]

      if check != 0:
        UP = cubeId
      else:
        LEFT = cubeId

    if lowerTagRead is not None:
      lowerTagHex = binascii.hexlify(lowerTagRead)
      check = allTags[lowerTagHex][1] % 2
      if check != 0:
        DOWN = cubeId
      else:
        RIGHT = cubeId
```

Come si può notare, ciascun lettore legge due facce e può distinguere su quale faccia si trova il cubo rilevato grazie alla disposizione dei tag e ai codici assegnati, effettuando un controllo sulla parità dei codici dei tag e quindi assegnando il codice del cubo rilevato alla variabile `RIGHT`, `UP`, `LEFT` o `DOWN`.

7.3 Motion sensing

L'implementazione di base per la lettura dei sensori presenti sull'IMU LSM9DS1 è stata effettuata mediante la libreria BerryIMU sviluppata da Mark Williams per OzzMaker [6]. Questa libreria si compone dei seguenti moduli:

- `LSM9DS1.py`: definisce i registri che vengono letti e/o scritti, così come da specifiche del produttore e gli indirizzi I2C del sensore (`0x1E` per il magnetometro e `0x6B` per l'accelerometro e il giroscopio).
- `IMU.py`: definisce i metodi per leggere e scrivere i dati nei registri del sensore, quindi sia per leggere i dati forniti dal sensore sia per configurare il sensore e

modificarne eventuali parametri.

Ad esempio i registri read-only contenenti i valori rilevati dal giroscopio lungo l'asse x vengono così definiti:

```
LSM9DS1_OUT_X_L_G  = 0x18   # registro contenente il primo byte
↪  del valore rilevato dal giroscopio
LSM9DS1_OUT_X_H_G = 0x19   # registro contenente il secondo byte
↪  del valore rilevato dal giroscopio
```

I valori dei sensori, su 16 bit, sono espressi in complemento a 2 e siccome i registri sono a 8 bit tali valori devono essere combinati. Il metodo che legge i registri relativi all'asse x del giroscopio e restituisce l'angolo di rotazione è così definito:

```
LSM9DS1_GYR_ADDRESS = 0x6B

def readGYRx():
gyr_l = bus.read_byte_data(LSM9DS1_GYR_ADDRESS, LSM9DS1_OUT_X_L_G)
  gyr_h = bus.read_byte_data(LSM9DS1_GYR_ADDRESS, LSM9DS1_OUT_X_H_G)

  gyr_combined = (gyr_l | gyr_h << 8)
  return gyr_combined if gyr_combined < 32768 else gyr_combined -
  ↪  65536
```

Il procedimento è del tutto analogo per gli altri assi e per i valori dell'accelerometro e del magnetometro; i valori che vengono restituiti dai metodi di lettura dei dati sono compresi tra -32.768 e 32.768.

I dati recuperati dai registri dei sensori presenti nel modulo LSM9DS1 devono essere opportunamente filtrati in modo da ridurre il rumore introdotto nelle singole misurazioni. A tal scopo vengono generalmente utilizzati i seguenti filtri: filtro passa-basso, filtro complementare o filtro di Kalman.
Il filtro passa-basso permette di filtrare le fluttuazioni di breve durata riducendo in questo modo l'impatto di quelle variazioni dovute al rumore e introdotte nella misura riportata. Questa tipologia di filtro trova ampio utilizzo per quanto riguarda l'accelerometro e il magnetometro.
Il seguente codice legge i valori dell'accelerazione lineare lungo i tre assi x, y, z ed applica un filtro passa-basso, caratterizzato dalla costante `LPF_FACTOR` che ne regola l'intervento:

```
LPF_FACTOR = 0.7
oldXAccRawValue = 0
```

```
oldYAccRawValue = 0
oldZAccRawValue = 0

while True:
aX = IMU.readaX()
aY = IMU.readaZ()
aZ = IMU.readaY()

  aX = aX * LPF_FACTOR + oldXAccRawValue * (1 - LPF_FACTOR)
  aY = aY * LPF_FACTOR + oldYAccRawValue * (1 - LPF_FACTOR)
  aZ = aZ * LPF_FACTOR + oldZAccRawValue * (1 - LPF_FACTOR)

  oldXAccRawValue = aX
  oldYAccRawValue = aY
  oldZAccRawValue = aZ
```

La problematica che affligge invece il giroscopio è di natura opposta: esso è infatti soggetto ad un fenomeno definito *drift*, ossia deriva, e consiste nell'accumulo di rumore nel tempo, che rende del tutto inutilizzabile il valore in uscita dal sensore. A tal scopo si utilizza generalmente un filtro passa-alto.

Utilizzo dell'IMU

Attualmente l'IMU viene utilizzata con i seguenti fini:

1. Attivazione dell'installazione museale: il sistema CuBee è inizialmente in stand-by e viene attivato mediante lo shake di uno dei cubi presenti sul piano di gioco, il quale invia un messaggio al computer che avvia la riproduzione del cortometraggio e predispone i cubi per il primo gioco. A tal fine quindi si è implementato una funzione che rileva tale gesture utilizzando l'accelerometro presente nel dispositivo.
2. Scelta all'interno dei giochi: è stata prevista la possibilità di effettuare alcune scelte relative alle dinamiche dei giochi mediante il *tilt* (inclinazione) del cubo lungo gli assi x e y combinando i dati dell'accelerometro con quelli forniti dal giroscopio.

Per quanto riguarda il primo punto, la rilevazione dello shake del cubo, una parte consistente del codice è stata sviluppata sulla base del codice open-source utilizzato dai dispositivi micro:bit sviluppati dalla BBC [7].
Con il termine shake, o scuotimento, si intende l'alternanza di un movimento in una direzione e un movimento nel verso opposto, ripetuto un certo numero di volte in un arco temporale molto breve, generalmente inferiore ai 2 secondi. Per rilevare

tale gesture si sfrutta l'accelerometro e al lato pratico si devono rilevare una serie di *zero-crossings*, ossia un'alternanza di accelerazioni positive e accelerazioni negative lungo un asse dell'accelerometro. Tale alternanza deve quindi verificarsi all'interno di una finestra temporale molto breve e il valore assoluto delle accelerazioni prese in considerazione deve essere superiore alla soglia definita in fase di programmazione, in modo da escludere false rilevazioni.

Il codice utilizzato per la rilevazione dello shake è il seguente:

```
import datetime, math, time, IMU

aXmax = 0
aYmax = 0

shakeX = False
shakeY = False

crossings = 0
timer = 0
timeout = 30
damping = 10

crossingsTHRS = 4
accTOL = 12
cubeShaken = False

while True:
  aX = IMU.readACCx()
  aY = IMU.readACCy()
  shakeDetected = False

  if (aX < -accTOL and shakeX) or (aX > accTOL and not(shakeX)):
    shakeDetected = True
    shakeX = not(shakeX)
    aXabs = math.fabs(aX)
    if aXabs > aXmax:
      aXmax = aXabs

  if (aY < -accTOL and shakeY) or (aY > accTOL and not(shakeY)):
    shakeDetected = True
    shakeY = not(shakeY)
    aYabs = math.fabs(aY)
    if aYabs > aYmax:
      aYmax = aYabs
```

```
if shakeDetected and crossings < crossingsTHRS:
  crossings += 1 # incrementa il conteggio dei zero-crossing

  if crossings == 1:  # se è il primo zero-crossing inizializza il
  ↪  timer
    timer = 0

  if crossings == crossingsTHRS:  # raggiunto il minimo n. di
  ↪  zero-crossings
    timer = 0
    cubeShaken = True
    if aXmax > aYmax:
      alongAxis = 0  # asse x
    else:
      alongAxis = 1  # asse y

    detectedShake(onAxis)

if crossings > 0:
  timer += 1

  if cubeShaken and timer >= timeout:
    cubeShaken = False
    timer = 0
    crossings = 0
```

Per quanto riguarda la rilevazione dell'inclinazione del dispositivo si deve tenere in considerazione che l'accelerometro fornisce una buona indicazione dell'inclinazione del dispositivo in condizioni statiche, mentre il giroscopio è adatto per eseguire tale misurazione in condizioni dinamiche. In questo progetto interessa rilevare l'angolo di inclinazione rispetto all'asse z, perpendicolare alla superficie del display, e l'asse lungo il quale il dispositivo viene inclinato, in modo da poter combinare tale angolo con le facce del cubo intersecate dagli assi x e y e avere a disposizione quattro possibili casi.
Per combinare le misure dell'accelerometro e del giroscopio e ottenere l'angolo θ di inclinazione del dispositivo si è utilizzato un filtro complementare che ha permesso di raggiungere una precisione ritenuta sufficiente, in considerazione dell'utilizzo previsto per i dispositivi.

Come si può notare nella figura 7.1 riportante lo schema del filtro complementare, l'angolo di inclinazione $\hat{\theta}$ relativo all'asse x è la somma dell'angolo θ ottenuto dai dati dell'accelerometro e sottoposto a filtraggio mediante un filtro passa-basso, e lo stesso angolo θ calcolato a partire dai dati del giroscopio e filtrato mediante un

filtro passa-alto.

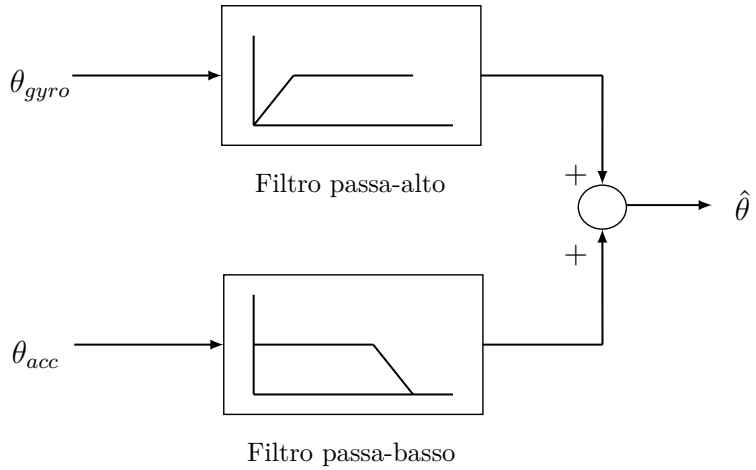


Figura 7.1: Schema di un filtro complementare

L'angolo di inclinazione $\hat{\psi}$, relativo all'asse y, è ottenuto in modo del tutto analogo. Il seguente modulo effettua la lettura dei dati relativi all'accelerometro e al giroscopio, filtrando e combinando i rispettivi angoli ottenuti mediante un filtro complementare caratterizzato dalla costante `CFA`:

```
import datetime, math, time, IMU

a = datetime.datetime.now()
G_GAIN = 0.07  # Sensibilità del giroscopio = 2000 dps
CFA = 0.45  # Costante del filtro complementare
RAD_TO_DEG = 57.29578
M_PI = 3.14159265358979323846

def estimateAngle(angle, angRate, dt, AccAngle)
  estAngle = CFA*(angle + angRate*dt) + (1 - CFA)*AccAngle
  return estAngle

while True:
  b = datetime.datetime.now() - a
  a = datetime.datetime.now()
  dt = b.microseconds / (1000000 * 1.0)

  # Lettura dei registri dell'accelerometro e del giroscopio
  aX = IMU.readaX()
  aY = IMU.readaY()
  aZ = IMU.readaZ()
```

```
    GYRx = IMU.readGYRx()
    GYRy = IMU.readGYRy()
    GYRz = IMU.readGYRz()

    # Conversione dell'angolo del giroscopio in gradi
    angXRate = GYRx * G_GAIN
    angYRate = GYRy * G_GAIN
    angZRate = GYRz * G_GAIN

    # Calcolo degli angoli di inclinazione dell'accelerometro, in
     ↪  gradi
    angleXAcc = math.atan2(aX, math.sqrt(aY*aY +aZ*aZ)) * RAD_TO_DEG
    angleYAcc = math.atan2(aY, math.sqrt(aX*aX + aZ*aZ)) * RAD_TO_DEG

    # Filtraggio mediante filtro complementare
    angleX = estimateAngle(angleX, angXRate, dt, angleXAcc)
    angleY = estimateAngle(angleY, angYRate, dt, angleYAcc)

    return angleX, angleY
```

Interessa inoltre studiare ulteriori gesture per migliorare l'interazione non solo con i singoli cubi, ma anche tra i cubi e il cortometraggio, e valutare anche modalità di gioco e ambiti diversi di applicazione dei cubi sfruttando tutti e tre i sensori presenti nei dispositivi. Tali future implementazioni saranno oggetto di studio successivo.

7.4 Illuminazione delle facce laterali

L'illuminazione delle facce laterali dei cubi è ottenuta mediante i LED WS2812, per la programmazione dei quali si è utilizzata la distribuzione Python della libreria rpi-ws281x sviluppata da Jeremy Garff [8].
Al pari di analoghi componenti hardware è necessario innanzitutto configurare i led ed impostare una serie di parametri; ciò avviene mediante le seguenti istruzioni:

```
import time, argparse
from neopixel import *

LED_COUNT  = 4  # numero totale dei led collegati
LED_PIN = 18  # pin BCM al quale sono connessi i led
LED_FREQ_HZ = 800000
LED_DMA = 10
LED_INVERT  = False
```

```
LED_BRIGHTNESS = 255
LED_CHANNEL = 0
```

La creazione dell'oggetto che rappresenta la striscia di led collegata al `LED_PIN` avviene tramite la classe `Adafruit_NeoPixel()`, alla quale vengono forniti i parametri precedentemente impostati:

```
ledStrip = Adafruit_NeoPixel(LED_COUNT, LED_PIN, LED_FREQ_HZ,
 ↪  LED_DMA, LED_INVERT, LED_BRIGHTNESS, LED_CHANNEL)
ledStrip.begin()
```

I colori sono rappresentati mediante l'oggetto `Color`, così definito: `Color(r, g, b)`, con `r`, `g` e `b` valori compresi tra `0` e `255` che rappresentano rispettivamente la componente rossa, verde e blu del colore che si vuole rappresentare.
Il colore di un singolo led, anche definito pixel, viene impostato attraverso il metodo `setPixelColor(j, color)`, con `j`, indice del led della striscia che si vuole accendere, compreso tra `0` e `LED_COUNT-1`, e `color` il colore che si vuole utilizzare. La luminosità dei led è regolabile, oltre che attraverso il parametro `LED_BRIGHTNESS` passato come argomento in fase di creazione dell'oggetto `ledStrip`, anche con il metodo `setBrightness(val)`, con `val` compreso tra `0` e `255`. Infine, per accendere i led della striscia, è necessario invocare il metodo `show()`.
Il codice completo per accendere di verde la striscia `ledStrip` precedentemente definita è quindi il seguente:

```
green = Color(0, 255, 0)
for j in range(ledStrip.numPixels()):
  ledStrip.setPixelColor(j, green)
ledStrip.show()
```

Impostato il colore dei singoli pixel della striscia e chiamato il metodo `show()`, la striscia manterrà tale impostazione indefinitamente, anche in caso di chiusura del programma in esecuzione sui dispositivi. Per aggiornare il colore dei led, questo dovrà quindi essere nuovamente impostato mediante `setPixelColor()` e infine reso *visibile* con `show()`. Non è previsto un metodo per spegnere i led e tale operazione viene effettuata impostando il valore `rgb(0, 0, 0)`, corrispondente al nero, come colore dei led.

I colori principali scelti per i diversi eventi possibili nell'esperienza dell'installazione museale sono i seguenti:

- Verde: segnala la buona riuscita di una determinata azione eseguita dall'utente o di una scelta corretta; ad esempio in *Unlock the cube* viene utilizzato il verde come feedback visivo ulteriore quando si trova la chiave giusta per sbloccare il cubo principale e visualizzarne il contenuto.
- Rosso: attraverso questo colore si segnala invece un evento dannoso o un pericolo all'interno dei videogiochi; ad esempio in *Ocean Getaway* i cubi attivi si illuminano di rosso quando l'utente esaurisce le bolle di ossigeno e quindi deve iniziare nuovamente la risalita in superficie.
- Giallo: si utilizza questo colore per richiedere l'attenzione dell'utente e far sì che quest'ultimo possa intervenire; ad esempio i cubi lampeggiano utilizzando il giallo quando l'utente deve scuotere i cubi per iniziare la riproduzione del cortometraggio e quindi dell'esperienza oppure per segnalare l'inizio dei giochi sui cubi, nei punti di interazione previsti.

In aggiunta a questi colori l'illuminazione delle facce laterali è correlata anche al mondo narrativo dei diversi giochi previsti, quindi i colori utilizzati sono molti di più, sebbene essi non siano universali nel proprio significato, come i tre appena citati.

L'illuminazione in *Unlock the cube*

Combinando le funzioni di gestione dei led elencate precedentemente, è possibile creare una serie di animazioni, in modo da rendere più *dinamico* il dispositivo. Ad esempio per quanto riguarda *Unlock the cube*, quando viene trovata la chiave corretta che permette di sbloccare il cubo, entrambi i cubi coinvolti nella connessione eseguiranno la seguente sequenza:

1. Impostazione del colore verde per ciascuno dei 4 led.
2. Accensione graduale fino ad arrivare alla massima luminosità dei led, `255`, completando l'operazione in circa 1 secondo.
3. Diminuzione graduale della luminosità dei led fino ad arrivare ad una luminosità pari a `175`, nell'arco di circa 1 secondo.
4. Aumento graduale della luminosità fino ad arrivare al valore massimo consentito nell'arco di circa 500 millisecondi.
5. Spegnimento graduale dei led in base alla durata del contenuto sbloccato sui cubi.

Per eseguire tale sequenza di operazioni si è definita la seguente funzione:

```
def fillFacets(color, fullBrightness, halfBrightness, count,
↪  lightFading, timer):

    t = fullBrightness - halfBrightness
    tUp = 1 / (2 * t)
    tDown = 1 / t
    tStart = 1 / 128
    i = 0

    for k in range(ledStrip.numPixels()):
        ledStrip.setPixelColor(k, color)

    for b in range(0, fullBrightness, 2):  # accensione dei led: 0
    ↪  -> fullBrightness
        ledStrip.setBrightness(g)
        ledStrip.show()
        time.sleep(tStart)

    while i < count:

        if i % 2 == 0:  # diminuzione luminosità: fullBrightness ->
        ↪  halfBrightness
            for c in range(fullBrightness, halfBrightness, -1):
                ledStrip.setBrightness(c)
                ledStrip.show()
                time.sleep(tDown)
        else:  # aumento luminosità: halfBrightness ->
        ↪  fullBrightness
            for c in range(halfBrightness, fullBrightness, 1):
                ledStrip.setBrightness(c)
                ledStrip.show()
                time.sleep(tUp)

    if lightFading:  # diminuzione luminosità in base alla durata
    ↪  del contenuto
        cFade = timer / fullBrightness
        for c in range(fullBrightness, 0, -1):
            ledStrip.setBrightness(c)
            ledStrip.show()
            time.sleep(cFade)
```

Per eseguire la sequenza di operazioni precedentemente descritta, la funzione appena riportata viene invocata con i seguenti argomenti: `fillFacets(green, 255, 175, 2, True, contentLength)`, con `green = Color (0, 255, 0)` e `contentLength`

la durata del contenuto visivo mostrato sui cubi.

7.5 Sviluppo di videogiochi con Pygame

I videogiochi sviluppati per questo progetto sono stati realizzati mediante la libreria Pygame, la quale fornisce una serie di funzioni necessarie a creare programmi con un'interfaccia grafica (GUI). Essa si basa sulla libreria Simple DirectMedia Layer (SDL), alla quale aggiunge altre funzioni, come le trasformazioni, l'applicazione di filtri o il meccanismo di collision detection, che la rendono più efficiente e preferibile a quest'ultima.

Struttura dei giochi

Il codice di un videogioco realizzato con Pygame può essere suddiviso in due parti principali: una parte di inizializzazione che viene eseguita una sola volta, al lancio dell'applicativo, e il game loop, ossia tutte quelle istruzioni che devono essere eseguite ciclicamente. Alcune istruzioni e funzioni contenute nella parte di inizializzazione per il cubo Main e relative al gioco *Unlock the cube* sono le seguenti:

```
import time, threading, datetime, binascii, sys, os, pygame
from pygame.locals import *
import Adafruit_PN532 as PN532
import OSC

os.putenv('SDL_FBDEV','/dev/fb1')  # utilizza il display SPI
 ↪  (/dev/fb1) come output
os.putenv('SDL_VIDEODRIVER', 'fbcon')  # forza l'output sul display
 ↪  SPI
os.putenv('SDL_MOUSEDRV', 'TSLIB')
os.putenv('SDL_MOUSEDEV', '/dev/input/event0')   # utilizza il
 ↪  touchscreen del display SPI

pygame.init()
pygame.mouse.set_visible(False)
clockObj = pygame.time.Clock()
FPS = 30

def startInit():
  # istruzioni di inizializzazione dei lettori NFC
  \ldots
```

```
  # configurazione del display
gameDisplay = pygame.display.set_mode((320, 240), 0)

# riproduzione della colonna sonora in loop
pygame.mixer.music.load('audio/bg_music1.mp3')
pygame.mixer.music.play(-1, 0.0)

# visualizzazione del lucchetto chiuso sul display
lockClosed = pygame.image.load('image/1_lock/i1.png').convert()
gameDisplay.fill(WHITE)
gameDisplay.blit(lockClosed, (0,0))
pygame.display.flip()
```

La frequenza di esecuzione del game loop è definita *frame rate* e generalmente è compresa tra i 15 e i 60 frame al secondo; ciò comporta che il computer deve essere in grado di generare la grafica da 15 a 60 volte al secondo. In assenza di controlli temporali sull'esecuzione del programma, il computer eseguirebbe le istruzioni di quest'ultimo alla massima velocità possibile. L'oggetto che permette di gestire in modo preciso l'intervallo tra i frame è `pygame.time.Clock`. Tale oggetto fornisce il metodo `tick(FPS)`, che viene eseguito come ultima istruzione del game loop.

Generazione della grafica

Dal punto di vista visivo il concetto di base di Pygame è dato dall'oggetto `Surface`, ossia un oggetto che rappresenta un'immagine rettangolare bidimensionale. Una serie di funzioni, dette *drawing*, permette di agire sui pixel di ciascuna `Surface` sia disegnando delle forme geometriche predefinite (linee, archi, poligoni, rettangoli, cerchi ecc.) sia disegnando ogni singolo pixel mediante i metodi offerti dall'oggetto `PixelArray`, il quale fornisce la rappresentazione matriciale di un oggetto `Surface`.

La creazione dell'oggetto display Surface, ossia la struttura contenente i diversi oggetti grafici che dovranno essere visualizzati sullo schermo, avviene mediante l'istruzione `gameDisplay = pygame.display.set_mode((width, height), 0)`. Qualunque disegno, immagine o variazione di pixel realizzati su una `Surface`, dovrà essere successivamente copiata in `gameDisplay` per poter essere visualizzata sullo schermo. La visualizzazione di un'immagine avviene quindi in tre passaggi:

1. `imageSurf = pygame.image.load(fileName)`: il modulo `pygame.image` determina automaticamente il tipo di immagine e crea un nuovo oggetto `Surface` contenente i dati dell'immagine.
2. `screen.blit(imageSurf, (x, y)`: caricata l'immagine come da passo precedente, per poterla rendere visibile sullo schermo è necessario copiare i pixel di

`imageSurf` sull'oggetto display Surface. La tupla `(x, y)` definisce la posizione a partire dalla quale verranno copiati i pixel di `imageSurf`. Il punto di riferimento per le coordinate è l'angolo in alto a sinistra dello schermo, il quale coincide con la posizione `(0, 0)`.

3. `pygame.display.flip()`: infine è necessario informare il sistema che è avvenuto un cambiamento per il quale si richiede l'aggiornamento dello schermo e quindi tutte le modifiche effettuate sulla display Surface verranno rese visibili sullo schermo.

È possibile disegnare molteplici contenuti sullo schermo, i quali verranno rappresentati in base all'ordine di *disegno* sulla superficie dello schermo, avendo quindi in primo piano l'ultimo oggetto disegnato mediante l'istruzione `screen.blit()`. Alla luce di queste considerazioni risulta quindi evidente che per animare un personaggio, per ciascun frame il game loop dovrà:

- aggiornare il personaggio e gli eventuali oggetti presenti, ad esempio cambiandone la posizione o le caratteristiche fisiche;
- eliminare dalla display Surface le tracce del personaggio e degli eventuali oggetti ri-disegnando il background;
- disegnare il personaggio e gli eventuali oggetti sullo schermo;
- aggiornare lo schermo.

Pygame permette l'utilizzo di formati quale il PNG che supportano l'utilizzo della trasparenza e fornisce inoltre una serie di metodi che permettono di agire sulla trasparenza dei singoli pixel (*pixel alpha*), di rendere trasparente i pixel di un determinato colore (*colorkey*) e di agire sulla trasparenza di un'intera `Surface` (*surface alpha*).

Sprite e animazioni

In aggiunta alle immagini, alle forme geometriche semplici come linee, rettangoli, cerchi ecc. e alla manipolazione diretta dei pixel dello schermo, un oggetto comunemente utilizzato nella programmazione di videogiochi è dato dallo *sprite*: si tratta di risorse grafiche, ossia immagini, che rappresentano personaggi e oggetti che si ritrovano nei giochi e, poiché vengono riutilizzate frequentemente, sono generalmente più efficienti da gestire. Il modulo che permette di creare e controllare gli sprite all'interno di Pygame è `pygame.sprite`; esso fornisce anche il meccanismo di collision detection, di primaria importanza nello sviluppo di un videogioco.
Tale modulo fornisce la classe `Sprite`, una classe di base per gli oggetti di gioco visibili, la classe `Group` che rappresenta un contenitore atto a gestire una serie di

oggetti `Sprite`, e infine vengono forniti diversi metodi relativi alla collision detection, che permettono di rilevare le collisioni che si possono verificare tra gli oggetti rappresentati dagli sprite.

Supponendo di dover realizzare un'animazione composta da 5 frame, a seconda dell'ordine in cui vengono mostrati tali frame si possono distinguere quattro tipologie di animazioni[9]:

1. Looping: l'animazione riprende dall'inizio una volta giunta alla fine, per cui la sequenza 0,1,2,3,4 viene eseguita indefinitamente 0,1,2,3,4,0,1,2,3,4,0,1, ...;
2. Forward: l'animazione viene eseguita ponendo i frame l'uno dopo l'altro, in ordine crescente, ossia 0,1,2,3,4;
3. Reverse: l'animazione viene eseguita a ritroso, invertendo l'ordine dei frame, ossia 4,3,2,1,0;
4. Ping-pong: questa sequenza prevede l'alternanza delle tipologie (2) e (3), per cui si avrà la sequenza 0,1,2,3,4,4,3,2,1,0.

Per realizzare queste animazioni è possibile utilizzare la seguente funzione, che riceve come parametri una lista con i nomi dei file delle immagini da utilizzare e un vettore con la sequenza da utilizzare per riordinare i frame e quindi modificare l'animazione risultante:

```
image_names = ["fr0.png","fr1.png","fr2.png","fr3.png","fr4.png"]
def get_sequence(frames_names, sequence):
  frames = []
  cache = {}
  for name in frames_names:
    if not cache.has_key(name):
      image = pygame.image.load(name)
      cache[name] = image
    frames.append(cache[name])

  animation_frames = []
  for k in sequence:
    animation_frames.append(frames[k])
  return animation_frames

sequence_looping = [0,1,2,3,4,4,3,2,1,0]
sequence_reverse = [4,3,2,1,0]
frames_looping = get_sequence(image_names, sequence_looping)
frames_reverse = get_sequence(image_names, sequence_reverse)
```

Infine l'animazione può essere visualizzata sullo schermo con un ciclo `for`, che ad intervalli regolari disegna un determinato frame e ne predispone la visualizzazione sullo schermo:

```
clockObj = pygame.time.Clock()
FPS = 30
while True:
    for f in frames_looping:
      gameDisplay.blit(f)
      pygame.display.flip()
      clockObj.tick(FPS)
```

Gestione degli eventi

Oltre ad aggiornare lo stato del gioco e disegnarne i componenti sullo schermo, il game loop gestisce anche gli eventi provenienti dai dispositivi di input del sistema e a tal fine utilizza il modulo `pygame.event`. I moduli che permettono l'utilizzo dei dispositivi di input all'interno del programma sono `pygame.mouse`, `pygame.key` e `pygame.joystick`. In questo caso ci interessa unicamente il modulo `pygame.mouse`, che permette, mediante il metodo `get_pos()`, di leggere le coordinate `(x, y)` del punto di tocco sul touchscreen e quindi di essere opportunamente gestite nel gioco per premere pulsanti, far muovere personaggi e oggetti, raccogliere bonus ecc. Gli eventi generati dal modulo `pygame.mouse` sono i seguenti:

- `pygame.MOUSEBUTTONDOWN`: generato quando il pulsante del mouse o il touchscreen viene premuto;
- `pygame.MOUSEBUTTONUP`: generato quando il dito viene sollevato dal pulsante del mouse o dal touchscreen;
- `pygame.MOUSEMOTION`: generato quando viene rilevato un movimento del mouse o sul touchscreen.

Parte III

Dimensione interattiva

Capitolo 8

Progettazione dell'interazione

— Elisa Musolino, Stelian Spulber

Nel presente capitolo si vuole porre l'attenzione sull'aspetto interattivo caratterizzante il sistema CuBee. Nello specifico verrà contestualizzato il progetto all'interno della dimensione narrativa tipica della tecnica di *storytelling interattivo*, con particolare riguardo agli spazi museali e al loro sviluppo odierno.
Successivamente verrà trattato il processo che ha dato origine alla struttura narrativa alla base del sistema e verranno riportate illustrazioni esplicative dei giochi realizzati per le parti di interazione effettive con l'utente. Infine verranno descritti i passaggi ed il software che hanno permesso l'integrazione finale del cortometraggio con i cinque dispositivi di gioco stampati in 3D.

8.1 Il contesto museale

Nel corso degli anni, il museo ha assunto ruoli diversi, partendo da tempio di ricerca scientifica e quindi luogo destinato ad ospitare artisti e letterati come il museion dell'Accademia platonica nell'antica Grecia, a luogo di conservazione e di contemplazione come le gallerie destinate al collezionismo da parte dei principi. Infine il museo diventa luogo di conservazione di beni culturali ma soprattutto un mezzo di comunicazione culturale, fondamentale per divulgare conoscenza, in particolar modo per le scuole, pertanto diventa *luogo di educazione e di diletto*.

Il museo, oltre a luogo di educazione, per poter essere efficace deve assumere anche la funzione di intrattenimento, una funzione che oggigiorno sta acquisendo sempre più rilevanza grazie all'introduzione di tecnologie che permettono ai visitatori di qualsiasi età di immergersi nei contesti creati, superando la fase in un certo modo passiva che limita l'utente ad osservare ciò che gli viene proposto.
Assume importanza non solo il concetto ma anche il modo in cui quest'ultimo viene comunicato, pertanto occorre prestare particolare cura alla tipologia di destinatari in modo da non porre questi ultimi nella situazione di inadeguatezza e far sì che tutti gli utenti possano comprendere i messaggi proposti.
Come riportato in [1], i musei possono usufruire di due categorie di comunicazione: le forme di *comunicazione indiretta* e quelle di *comunicazione in presenza*. Il primo tipo di comunicazione si riferisce alle situazioni in cui il mittente e il destinatario non possono interagire l'uno con l'altro, ad esempio nel caso in cui il visitatore del museo può solamente leggere le indicazioni preparate senza poter chiedere ulteriori spiegazioni, come i pannelli disposti nelle sale dei musei o le didascalie che descrivono le opere esposte. È quindi necessario che il linguaggio di comunicazione non sia un fattore limitante, ma piuttosto un elemento fondamentale per poter interagire con un pubblico di qualsiasi età, considerando sempre il contesto e il tipo di messaggio che si intende trasmettere. Come evidenziato da Mattozzi, la "semplificazione" delle informazioni non è la via corretta per poter trasformare delle nozioni in un linguaggio idoneo ad essere compreso a tutti, ma consiste essenzialmente "in quel processo di trasformazione dei saperi esperti in saperi scolastici, in modo da rendere percorribile ai diversi visitatori l'itinerario d'indagine".
D'altra parte, la comunicazione in presenza prevede l'intervento di entrambe le figure, l'utente e la guida, e consente l'interazione immediata tra visitatore e oggetto favorendo la comprensione e l'apprendimento. Fra queste forme di comunicazione vi è ad esempio la presenza di laboratori didattici che insieme alle nuove tecnologie consentono all'utente un accesso diretto ai contenuti proposti e grazie anche alla manipolazione degli oggetti, l'utente viene stimolato continuamente sia emotivamente sia acquisendo nozioni e creando nuovi messaggi.

> "I musei sono molto più che repository di oggetti; sono luoghi di incontro per persone e idee."
>
> – *Robert Hewison*

In un articolo pubblicato su *The Guardian* nel 2015 [2], la direttrice della Whitworth Art Gallery, Maria Balshaw, dà una propria opinione su come dovrebbero essere i musei di oggi, nel 2020:

> "I musei del futuro dovrebbero essere luoghi in cui le persone si sentono a proprio agio − per incontrare cose che potrebbero non sapere e cose che fanno. Dovrebbero essere luoghi in cui mescolarsi ed esplorare le cose in compagnia di estranei.

> Al loro interno, le persone dovrebbero trovare il passato, il futuro ed essere in grado di portare le proprie idee e apprenderne di nuove. I musei dovrebbero essere divertenti, curiosi, permetterci di vedere la bellezza e riempirci di meraviglia. Dovrebbero essere spazi socievoli, che annullano silenziosamente la gerarchia sociale e la disuguaglianza."

Nello scenario attuale, i musei tendono ad adottare una tecnica di comunicazione sempre più centrata sul pubblico, *audience oriented*, ossia i contenuti che vengono realizzati vertono sulle motivazioni e sulle aspettative degli utenti interessati. Infatti, analizzando gli utenti e sulla base dei loro feedback è possibile progettare contenuti che si adattano alle loro preferenze di intrattenimento, contribuendo così al raggiungimento delle necessità di entrambe le parti: l'ampliamento del pubblico e la propensione a partecipare ad un'esperienza museale di apprendimento e intrattenimento, e d'altra parte il coinvolgimento del pubblico stesso.

Come emerge dal report *Musei e Social media* del 2019 [3], fra gli elementi che si affermano a favore di una più efficace metodologia comunicativa da parte dei musei, vi è la prevalenza dei contenuti visuali che ottengono un riscontro doppio rispetto ad altri tipi di formati. Qualsiasi video, dal meno accurato a quello tecnicamente più sofisticato, è in grado di generare un'interazione con il pubblico.
Un ulteriore elemento che emerge fra le caratteristiche di un museo riguarda le motivazioni che vengono trasmesse mediante i contenuti proposti, pertanto è fondamentale quella che viene definita come *curation*, ossia l'attenzione che si pone nella selezione degli argomenti da trattare e il modo in cui questi vengono elaborati. Tali tematiche devono essere quindi sviluppate in modo da educare gli utenti attraverso spiegazioni e illustrazioni che attirano il loro interesse; anche se i contenuti richiedono una minima concentrazione, tale sforzo può essere accettato dall'utente purché il modo in cui vengono trasmessi i contenuti siano realmente efficaci.
Il terzo elemento significativo consiste nel rapporto che si crea tra l'utente e il museo, ossia l'utente necessita di essere un soggetto attivo all'interno dei contenuti proposti nel museo, e tale approccio trova un favorevole riscontro ad esempio nei casi degli *User-generated Content*, ovvero contenuti prodotti dagli utenti stessi e condivisi sui social; una strategia che alcuni musei adottano per ottenere un maggior coinvolgimento. Pertanto, oggigiorno l'utente non è più soltanto uno spettatore ma diventa co-produttore e quindi partecipe della creazione dei servizi legati all'espansione della cultura in ambito museale.

Il sistema CuBee si può inserire in quello che nel 1973 il documentarista per la National Geographic, Bob Heyman, definì *edutainment*, ossia una forma di comunicazione caratterizzata dalla combinazione di tre elementi: divertimento, intrattenimento ed emozione. Il termine edutainment venne utilizzato già a partire dagli anni '60 dal sociologo Marshall McLuhan, il quale affermò:

> "Coloro che fanno distinzione fra intrattenimento ed educazione forse non sanno che l'educazione deve essere divertente ed il divertimento deve essere educativo."

L'edutainment rappresenta due dei principali obiettivi della comunicazione culturale: l'education, ossia la fase educativa e di apprendimento, e l'entertainment che indica la fase di divertimento e di svago.
Lo sviluppo di nuovi mezzi tecnologici ha permesso di applicare tale concetto a diverse forme di intrattenimento utili all'acquisizione di conoscenza, andando ad inserirsi in quelli che vengono definiti luoghi della cultura, ossia quelle strutture in grado di conservare e allo stesso tempo di comunicare e diffondere cultura; tra questi vi sono i siti archeologici, i teatri, le pinacoteche, gli archivi, le biblioteche e i musei.
Le esperienze che si inseriscono all'interno dell'edutainment hanno quindi lo scopo di comunicare divertendo e si rivolgono ad un ampio target. Infatti si possono distinguere almeno tre tipi di edutainment/utenti corrispondenti a tre fasce di età [4]:

1. Edutainment in età prescolare: indica il primo approccio alle tecnologie da parte dei bambini, i quali vengono stimolati a compiere delle semplici scelte predefinite di carattere educativo.

2. Edutainment per ragazzi a diversi livelli scolari: le attività didattiche vengono affiancate e integrate da prodotti software.

3. Edutainment per un pubblico extrascolastico: offre approfondimenti visivi di cultura generale grazie ad esperienze che avvengono in determinati spazi e tramite tecnologie innovative, interattive e virtuali.

Quest'ultima forma di edutainment è quella a cui occorre rivolgersi principalmente in casi come i contesti museali in cui si ha la possibilità di coniugare attività interattive con quelle intercreative.

Esistono molti esempi di nuove tipologie di comunicazione che puntano a creare dei percorsi immersivi stimolando le emozioni; fra questi vi è *Peopling the Palaces*, un'installazione di Peter Greenaway realizzata nel 2007 presso il Palazzo Reale di Venaria. Si tratta di un'installazione multimediale basata sulla proiezione di un film sulle pareti del palazzo o su lastre di plexiglass appese al soffitto, figura 8.1; il video mostra cinque ambientazioni differenti in cui gli attori interpretano la vita ai tempi della dinastia dei Savoia e tramite luci e suoni accompagnano il visitatore durante il percorso.

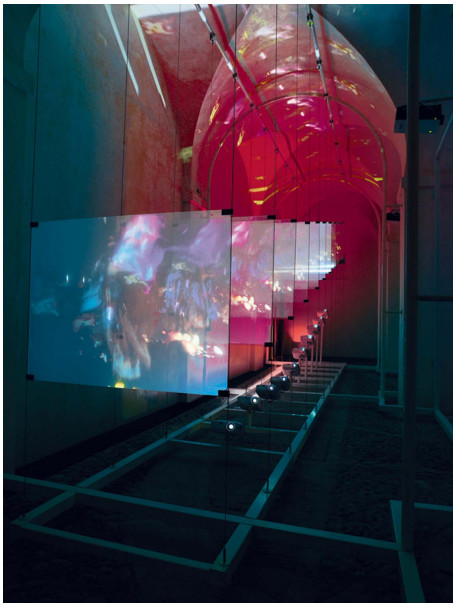


Figura 8.1: *Peopling the Palaces.* Fonte: Fabel Design.

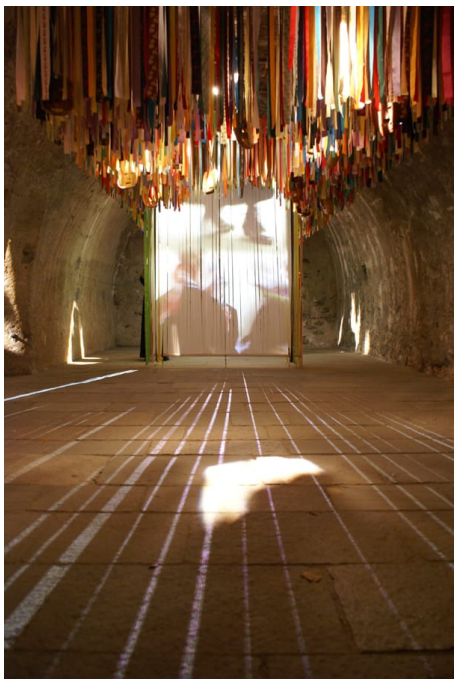


Figura 8.2: *La Montagna in Movimento*. Fonte: Alessandro Mellano.

Un ulteriore esempio è *La Montagna in Movimento*, un allestimento all'interno della fortificazione Forte Albertino a Vinadio. Si tratta di un'installazione in cui i muri diventano spazi narrativi rappresentati mediante proiezioni, postazioni desktop, telecamere a infrarossi, lastre ologrammatiche e schermi sensibili, attraverso cui vengono mostrate informazioni e contenuti didattico-formativi, figura 8.2.

Il progetto CuBee vuole essere inserito nell'ambito di un'educazione *informale* e non coercitiva, presente nella vita quotidiana, nei luoghi in cui le persone sono propense a raggrupparsi, come centri commerciali, piazze pubbliche, parchi e musei. La nostra scelta è ricaduta proprio su questi ultimi, i musei, contesti che permettono di creare percorsi di informazioni da adottare in diverse forme. In particolare, per la realizzazione effettiva dell'esperienza, considerata la tematica affrontata, si è scelto il Museo A come Ambiente di Torino.

Museo A come Ambiente

Il Museo A come Ambiente, MAcA, come suggerito dal nome, è un museo dedicato interamente a tematiche ambientali e si definisce come l'unico museo ambientale in Europa. Il museo è nato nel 2007 e conta un numero importante di visitatori, di cui le famiglie con bambini e adolescenti ne costituiscono una parte importante.

Pertanto questa tesi si è concentrata principalmente sulla progettazione e sullo sviluppo di un'interfaccia tangibile e in secondo luogo si è voluto applicare tale sistema in uno specifico contesto individuato, come quello museale, al fine di valutarne l'efficacia e verificare i possibili sviluppi e implementazioni del sistema anche in contesti e utilizzi differenti.
Per quanto riguarda la progettazione del dispositivo innanzitutto ci siamo focalizzati sui bisogni degli utenti tipici del MAcA, costituiti da un pubblico eterogeneo, a partire da bambini di 3 anni fino ad arrivare agli adolescenti, giovani genitori e nonni. Il nostro pubblico di riferimento è costituito soprattutto da bambini, 11-14 anni, e adolescenti, 14-18 anni.

Al fine dell'installazione organizzata all'interno degli spazi del Museo A come Ambiente, inizialmente sono state approfondite le installazioni proposte dal museo in uno specifico giorno di apertura, il giorno 16 novembre 2019. Il museo è suddiviso in cinque zone tematiche, all'interno delle quali sono previste una serie di installazioni permanenti che risultano adatte soprattutto per i più piccoli e per la fascia di pubblico di nostro interesse: Alimentazione, Acqua, Energia e Clima, Scarti e Trasporti.

Una delle installazioni presenti nella zona Alimentazione è *Il cibo divertente*, che riprende l'aspetto di una cucina e consiste nel realizzare una ricetta con la guida di uno chef. In tale spazio, mostrato in figura 8.3, sono presenti:

- una serie di ingredienti, acqua, sale, farina, limoni ecc., ognuno identificato dalla forma dell'ingrediente stesso o da un contenitore con un'etichetta identificativa;
- un cesto all'interno del quale devono essere posizionati gli ingredienti;
- un monitor frontale sul quale è presente lo chef che fornisce le indicazioni per realizzare la ricetta;
- un display touchscreen posizionato sul piano di lavoro della cucina, che fornisce le indicazioni testuali per l'esecuzione della ricetta.

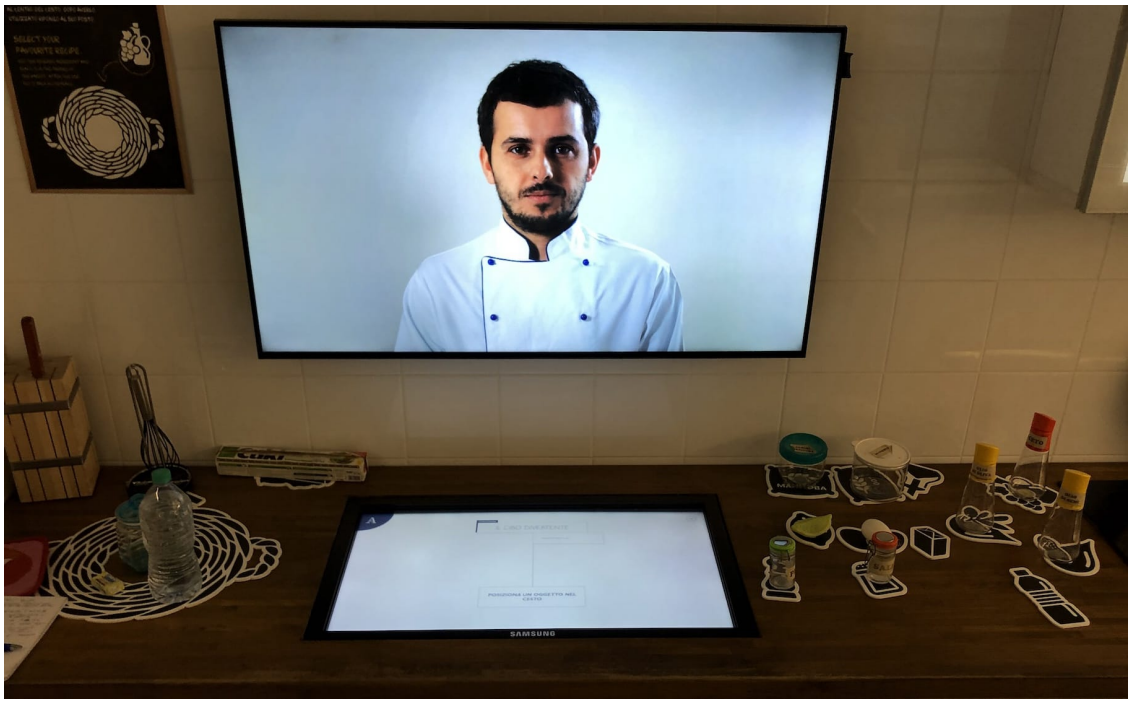


Figura 8.3: L'installazione *Il cibo divertente* all'interno della zona tematica Alimentazione del museo.

Dal punto di vista del funzionamento dell'installazione ogni ingrediente dispone di un tag NFC, mentre sotto il cartonato del cesto presente sul piano di lavoro vi è un lettore NFC che legge i tag posizionati, quindi gli ingredienti che vengono posizionati nel cestino. Alle indicazioni dello chef corrisponde un diverso video per ciascun passo della ricetta.
Si tratta chiaramente di un'installazione rivolta a bambini più piccoli rispetto al pubblico individuato per il nostro progetto, e prevede una modalità di interazione guidata.

Per quanto riguarda la zona tematica Acqua, una delle installazioni presenti è relativa ai mari e agli oceani e consiste nella riproduzione di una serie di brevi documentari a scelta dell'utente, proiettati su una superficie semicircolare, come si può notare nella figura 8.4. I documentari illustrano alcune delle caratteristiche relative alla vita marina presente a diverse profondità, con una voce guida a supporto delle immagini. La scelta del video da riprodurre avviene mediante i tasti di un citofono sul quale sono indicate le profondità cui si riferiscono i documentari relativi ai rispettivi pulsanti. La proiezione avviene mediante tre proiettori opportunamente orientati in modo da coprire i 180° della superficie di riproduzione.

Figura 8.4: Un'installazione della zona tematica Acqua del museo. A destra il citofono attraverso il quale si scelgono i documentari da riprodurre.

Oltre alle installazioni previste nelle diverse zone tematiche del museo, nel periodo interessato è stata organizzata la mostra dal titolo *Verso il futuro*, attraverso la quale si sono voluti raccontare gli obiettivi di sviluppo sostenibile fissati dall'Organizzazione delle Nazioni Unite per il 2030. Lo scopo della mostra è stato non solo quello di mettere in mostra una serie di problematiche ancora attuali e lontane da tali obiettivi di sostenibilità, ma soprattutto fornire anche alcune soluzioni ed invitare i cittadini ad assumere uno stile di vita più sostenibile e rispettoso dell'ambiente e della collettività, mettendo in atto un comportamento, come definito dai curatori della mostra, da "eco-eroe".

La mostra è stata pensata per essere guidata dal personale del museo, i divulgatori scientifici; tale approccio tuttavia può risultare poco efficace, in quanto all'utente non viene data del tutto la possibilità di scoprire e apprendere autonomamente alcuni concetti che si concretizzano nelle installazioni proposte, come quelle mostrate in figura 8.5.

Figura 8.5: A sinistra un'installazione interattiva sulla distribuzione dell'acqua delle piogge. A destra un'installazione che permette di visualizzare su una sfera rotante una serie di animazioni relative a dati climatici; quest'ultima installazione tuttavia non è adoperabile dall'utente ma solo dal personale del museo.

Durante la visita ci si è concentrati principalmente sull'osservazione dei visitatori e sulle loro interazioni con alcune delle installazioni proposte. I bambini, individuati come pubblico di riferimento principale del nostro sistema, sono stati quelli maggiormente d'interesse. Durante la giornata di osservazione nel museo i visitatori presenti erano esclusivamente famiglie, composte da madre, padre e figli ma anche figli accompagnati da un solo genitore. I bambini presenti avevano un'età compresa tra i 4 e i 14 anni.
Quanto si è notato conferma il fatto che sia per i genitori che per i bambini destano un interesse maggiore le installazioni che prevedono la manipolazione di oggetti, quindi che dispongono di una fisicità intrinseca; tuttavia, come da aspettative, per quanto riguarda i bambini, soprattutto sotto i 7-8 anni prevale la manipolazione di oggetti pressoché fine a se stessa, quindi ad esempio nel caso della proiezione dei documentari si è notato che per quattro diversi bambini interessava soprattutto il gesto di premere il pulsante del citofono che scatenava un'azione sullo schermo a loro visibile; per la durata della proiezione dei video, la loro attenzione era estremamente ridotta, dando più che altro l'impressione di attesa per poter premere nuovamente il pulsante ed eseguire una nuova azione. Questo comportamento non si è verificato per l'unico bambino di 14 anni presente il giorno della visita.

Per quanto riguarda l'interazione con i display touchscreen, si è riscontrata ancora una certa difficoltà nell'utilizzo di determinate applicazioni, in particolar modo per quanto riguarda l'interfaccia grafica. Ad esempio, nel caso di un gioco informativo proposto in un'installazione della zona tematica sull'acqua, figura 8.6, si è notato sia per i bambini che per gli adulti la difficoltà ad interagire con il display nonché con i contenuti ivi presenti, per via dei pulsanti poco evidenti; emblematico è il caso di un padre e di un bambino di 7 anni che hanno abbandonato la postazione affermando che l'installazione "non funziona".
Questi aspetti possono essere causati da svariati motivi ma il principale può essere attribuito ad una serie di mancanze nella fase di progettazione dell'esperienza, includendo una mancata valutazione con utenti appartenenti al pubblico di riferimento del contesto considerato.

Figura 8.6: Madre e figlio giocano su un touchscreen.

I visitatori del museo sembrano tuttavia apprezzare molte delle installazioni presenti nel museo e la visita viene vissuta anche come un "momento didattico", assimilando una serie di nozioni, come ad esempio sulla raccolta differenziata, non solo da parte dei bambini ma anche dai genitori.
Alla fine della visita, gli utenti si sentono più "informati e consapevoli" delle proprie scelte, ed attraverso la manipolazione degli oggetti e gli esperimenti scientifici l'esperienza museale rimane estremamente positiva anche per i più piccoli.

8.2 Storytelling interattivo

Come descritto nel capitolo 2, lo storytelling si può definire come una tecnica narrativa in grado di trasmettere una storia attraverso rappresentazioni testuali, visive e percettive mediante diversi canali comunicativi, e trova un impiego sempre più diffuso nel campo educational.

La combinazione tra narrazione e tecnologia dà luogo alla forma di narrazione definita come *digital storytelling* che prevede la costruzione di una narrazione visualizzabile tramite diversi strumenti di espressione digitale. L'elemento distintivo che rende interattivo lo storytelling consiste nella creazione di storie basate sulla partecipazione e sulla collaborazione dell'utente con il sistema digitale.

Pertanto, la differenza tra le storie tradizionali e le storie interattive è essenzialmente la possibilità da parte degli spettatori di approcciarsi in maniera diretta a queste ultime e utilizzarle per personalizzare e condividere i contenuti multimediali. La progettazione di una storia interattiva richiede un diverso modo di pensare rispetto alla classica storia, in quanto occorre creare dei collegamenti logici efficaci per poter articolare una storia manipolabile dall'utente interessato.
Occorre innanzitutto selezionare le informazioni principali riguardo al tema che si intende trattare e successivamente organizzare tali risorse per inserirle in un contesto che abbia un significato logico in grado di dare vita ad un percorso di apprendimento usufruibile dall'utente destinatario. All'utente vengono quindi proposti una serie di compiti che deve svolgere, in parte in maniera guidata ed in parte utilizzando le proprie conoscenze e abilità, per portare a termine degli obiettivi ben definiti. È quindi necessario che in ciascuna narrazione venga bilanciata la libertà di interazione dell'utente con il sistema, in quanto un'ampia libertà di scelta concessa all'utente renderà difficile il compito da parte del sistema il quale non riuscirà a controllare adeguatamente lo sviluppo della storia; d'altra parte un'interazione limitata risulterà poco coinvolgente e quindi generalmente poco apprezzata dall'utente.

Esistono due distinzioni di storytelling: *crossmedia storytelling* e *transmedia storytelling*. Nel primo caso, vi è un'unica storia che viene veicolata attraverso diversi canali di comunicazione lasciando invariato il contenuto ma viene solamente adattato alle caratteristiche e al contesto di ciascun mezzo utilizzato. Ad esempio un unico racconto può essere visto in un film, letto in un libro oppure può essere trasmesso sotto forma di videogioco.

Questo nuovo approccio alla narrazione di una storia di tipo crossmediale fa riferimento al concetto definito dal filologo russo Boris Viktorovič Tomaševskij, il quale ideò il concetto per cui un testo narrativo può essere destrutturato in unità più piccole. Da qui la distinzione tra *fabula* e *intreccio*; nello specifico esistono tre possibilità di comunicare una storia su diversi medium [5]:

1. Stessa fabula e stesso intreccio: si tratta del remake, ossia di ricreare una stessa storia su un medium che usa un linguaggio diverso ma riproducendo il contenuto con precisione per ottenere un risultato più possibile uguale all'originale.

2. Stessa fabula ma intreccio diverso: in questo caso ci si riferisce al *retelling* e prevede la trasmissione della storia invariata ma cambia il modo in cui viene raccontata, sfruttando le caratteristiche dei diversi medium.

3. Diversa fabula e diverso intreccio: si tratta del *reimagining* che prevede la modifica della storia e delle ambientazioni e nel passaggio da un medium all'altro

si creano contenuti diversi mantenendo però delle caratteristiche della storia originale.

D'altra parte, lo storytelling di tipo transmediale è una forma di comunicazione che separa una storia in più parti con diversi contenuti fruibili su diversi mezzi di comunicazione come film, serie tv, videogiochi, fumetti, e solamente tramite l'interazione e il contributo degli utenti è possibile ricostruire l'intera storia. Quest'ultima deve essere strutturata in modo efficace e deve essere facilmente comprensibile dall'utente per poterla ricostruire. Nel libro *Cultura convergente* [6], Henry Jenkins definisce lo storytelling transmediale come "un processo nel quale gli elementi integrati di una narrazione vengono sistematicamente separati e diffusi tramite diversi canali di comunicazione, con lo scopo di creare una esperienza di intrattenimento unificata e coordinata. Idealmente, ogni media dà un contributo unico allo sviluppo della storia".

Nel panorama odierno dello storytelling interattivo si ritrovano diversi esempi che prevedono unicamente l'uso di puntatori e il web per far interagire l'utente con i dispositivi tecnologici. Tra questi vi si trova il più recente esperimento lanciato a fine 2018 sulla piattaforma Netflix, intitolato *Bandersnatch*. Si tratta di un film della nota serie *Black Mirror*, caratterizzato da una narrazione di tipo interattiva. In alcuni punti la narrazione viene interrotta e compare sullo schermo del dispositivo una grafica che presenta due opzioni testuali che l'utente può scegliere tramite il puntatore del computer o con il telecomando, come mostrato in figura 8.7. In base a tale scelta la storia proseguirà in maniera diversa. Si ha inoltre la possibilità di andare avanti e indietro nel percorso narrativo per esplorare le diverse scelte.

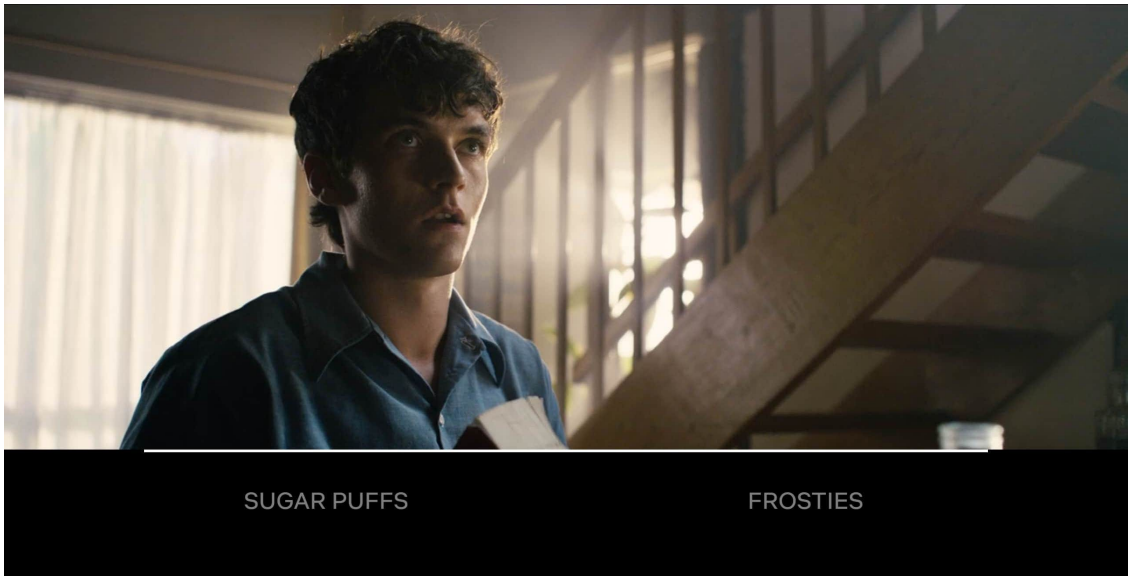


Figura 8.7: Frame del film *Black Mirror Bandersnatch*. L'utente deve scegliere una delle due opzioni proposte, entro il tempo indicato dalla barra bianca. Fonte: Netflix.

L'intera storia è stata scritta e progettata con l'ausilio della piattaforma open source Twine[7], che consente di realizzare storie interattive e non lineari.

Un ulteriore progetto interattivo è *Oat the Goat*, una storia sviluppata dalla compagnia di produzione Assembly nel 2018. Nello specifico, si tratta di un racconto di animazione suddiviso in undici scene della durata complessiva di dodici minuti realizzato nell'ambito della settimana dedicata al bullismo in Nuova Zelanda. È disponibile gratuitamente online e si pone l'obiettivo di insegnare ai bambini quali scelte compiere in diverse situazioni di difficoltà. Presenta inoltre un voice-over che narra la storia all'utente e lo guida nelle opzioni di scelta, illustrate sullo schermo del dispositivo usato. Un esempio di scelte proposte durante la narrazione è visibile in figura 8.8; sono presenti quattro carte, ciascuna con un'opzione diversa che chiede all'utente il comportamento da mettere in atto in un determinato momento della storia. La scelta viene effettuata tramite un click su una carta alla volta; in base alla carta scelta viene mostrata una scena diversa e la storia prosegue solamente quando viene scelta la carta corretta.

Figura 8.8: Screenshot tratto dal progetto interattivo *Oat the Goat*.

Nella prima fase di progettazione del sistema CuBee, l'idea è stata quella di creare una storia interattiva basata sull'approccio di interazione presente negli esempi appena citati.
Si è inizialmente effettuata una prova utilizzando il software Ren'Py [8], usato nella creazione di visual novel interattive tramite il computer. Si tratta di un programma che permette di creare storie ramificate e presenta una sintassi simile ad una sceneggiatura. Ren'Py è basato su Pygame e quindi usa Python, permettendo anche la scrittura di blocchi di codice per l'implementazione di funzioni personalizzate. È

stata quindi implementata una narrazione per simulare il viaggio della protagonista Pinnie mettendo a disposizione dell'utente delle opzioni di scelta durante il percorso; in base a tali scelte lo scenario cambiava e la storia proseguiva. Tale approccio, tuttavia, risultava limitante, in quanto l'interazione sarebbe potuta avvenire utilizzando un solo dispositivo di gioco come sostituto di un controller per effettuare le scelte mostrate su uno schermo, rendendo forzato l'uso dei restanti quattro cubi. Inoltre non permetteva di sfruttare al massimo l'aspetto interattivo. In figura 8.9 è riportato uno screenshot della prova effettuata su Ren'Py in cui sono state programmate due opzioni che consentono all'utente di scegliere l'ambiente in cui proseguire la narrazione, Artide od Oceano. In questo caso l'opzione scelta veniva evidenziata di giallo per confermare l'azione dell'utente. Selezionato l'ambiente, lo scenario cambiava mostrando all'utente ulteriori scelte e azioni da compiere.

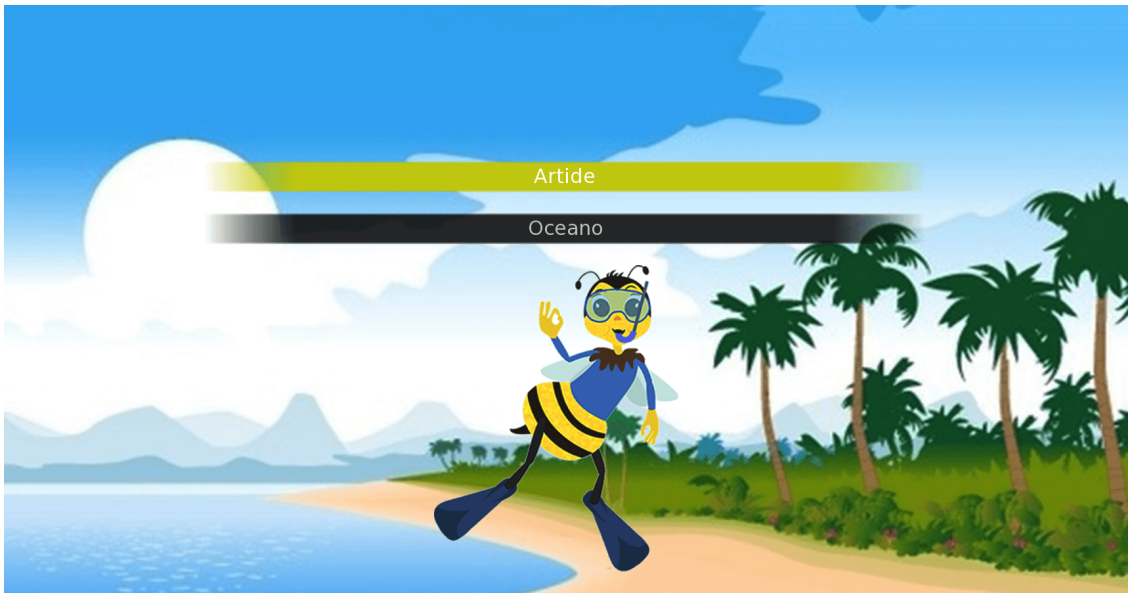


Figura 8.9: Grafica con la prima versione di Pinnie disegnata in modalità *flat* e uno sfondo di prova usato come reference.

CuBee è stato progettato con l'obiettivo di creare un sistema di storytelling interattivo, partendo da una storia di base che si articola in diverse parti separate, la cui combinazione dà origine ad evoluzioni diverse della storia cardine.
In seguito allo sviluppo dei concetti chiave della narrazione è stata effettuata la scelta dei dispositivi su cui veicolare la storia considerando anche il contesto museale e quindi lo spazio disponibile. Si è quindi deciso di trasmettere il contenuto su due tipologie di schermi: un proiettore e gli schermi dei cinque dispositivi cubici. La struttura narrativa e il set di decisioni progettate verranno descritte in dettaglio nel paragrafo successivo.

La volontà di creare nuove forme di narrazione non più lineari, deriva dalla necessità di sperimentare le nuove tecnologie che continuano ad aumentare la connessione tra le persone. Oggigiorno gli utenti vengono sempre più attratti da contenuti che permettono di essere manipolati, personalizzati e condivisi, e l'interazione rappresenta probabilmente il miglior modello di comunicazione attraverso cui informare gli utenti. I nuovi autori di storie interattive devono quindi essere in grado di anticipare le reazioni degli utenti di fronte a una scelta, "pensare quindi in seconda persona" come afferma Chris Crawford.

8.3 Design dell'esperienza

Il design dell'esperienza, ossia il modo in cui viene progettata un'esperienza, influisce sul riscontro che avranno gli utenti coinvolti e tale modalità può essere utilizzata in *presenza* o in *remoto*. In [4], gli autori usano il concetto di caldo o freddo per descrivere queste due modalità di progettazione, ma dando un significato diverso da quello associato da Marshall McLuhan ne *Gli strumenti del comunicare*, in cui egli effettua una distinzione tra *media caldi* come la radio o il cinema e *media freddi* come il telefono o la tv. Mario Ricciardi, nel suo libro *La comunicazione. Maestri e paradigmi*, riporta la definizione di McLuhan del concetto di caldo e freddo e la distinzione tra i due [9]:

> "È caldo il medium che estende un unico senso fino a un'«alta definizione»: fino allo stato, cioè, in cui si è abbondantemente colmi di dati. Dal punto di vista visivo, una fotografia è un fattore di «alta definizione», mentre un *cartoon* comporta una «bassa definizione», in quanto contiene una quantità limitata di informazioni visive. Il telefono è un *medium* freddo, o a bassa definizione, perché attraverso l'orecchio si riceve una scarsa quantità di informazioni, e altrettanto dicasi, ovviamente, di ogni espressione orale rientrante nel discorso in genere perché offre poco ed esige un grosso contributo da parte dell'ascoltatore. Viceversa i *media* caldi non lasciano molto spazio che il pubblico debba colmare o completare; comportano perciò una limitata partecipazione, mentre i media freddi implicano un alto grado di partecipazione o di completamento da parte del pubblico. È naturale quindi che un *medium* caldo come la radio abbia sull'utente effetti molto diversi da quelli di un *medium* freddo come il telefono. [...] Un *medium* caldo permette meno partecipazione di un *medium* freddo."

Tali termini vengono qui usati per indicare invece la capacità di coinvolgimento emozionale del dispositivo tecnologico presentato nei confronti dell'utente.
In questo caso, la modalità in presenza viene definita come modalità *calda* per indicare una capacità di comprensione del messaggio superiore da parte dello spettatore in quanto quest'ultimo si trova a diretto contatto con l'oggetto ed ha quindi la possibilità di interagire ed integrare la propria esperienza con contenuti aggiuntivi.
La modalità in remoto consiste invece in un'esperienza di tipo *fredda* in quanto

consente all'utente che non si trova fisicamente nell'ambiente in cui si svolge l'esperienza di entrare in contatto con l'oggetto tramite la progettazione di ambienti virtuali.

Il tipo di comunicazione che si è voluto realizzare per il sistema CuBee si può ricondurre alla modalità in presenza in quanto presuppone una partecipazione attiva e diretta tra l'utente e il sistema. Si è quindi voluto dare la possibilità all'utente di contribuire in prima persona all'esperienza, compiendo scelte che portano all'evolversi di situazioni diverse.
Pertanto, i contenuti realizzati vengono proposti all'utente con l'obiettivo di creare un percorso immersivo unendo l'aspetto coinvolgente offerto dalla tecnologia, in particolare tramite i giochi, con l'aspetto informativo relativo alla sfera didattica.

8.3.1 Struttura della narrazione

Il sistema interattivo CuBee si basa su un tipo di narrazione spaziale in quanto la storia viene distribuita sotto forma di contenuti diversi, dalla dimensione puramente visiva data dalla proiezione del cortometraggio alla dimensione, oltre che visiva, tangibile e interattiva attraverso cui il racconto si sviluppa nella modalità di videogioco. Quest'ultimo aspetto verrà trattato in dettaglio nel paragrafo 8.4.

CuBee prevede l'integrazione di due attori principali: da una parte la sfera delle specie animali e dall'altra gli utenti coinvolti nella narrazione, la cosiddetta sfera umana. Nello specifico, per quanto riguarda gli animali, la scelta della figura maggiormente rappresentativa del regno animale è ricaduta sull'ape, la quale è considerata essere tra i principali indicatori biologici in grado di valutare lo stato di salute ambientale di un particolare ecosistema e i rischi di esposizione degli animali.
Le api rappresentano quindi gli elementi da cui dipendono tutti gli animali e di conseguenza anche l'uomo. Da qui è scaturita la scelta di definire un cosiddetto ecosistema delle api e utilizzarlo come elemento portante dell'intera narrazione sviluppata.

Dal punto di vista narrativo il cubo rappresenta il mezzo attraverso cui le api, in particolar modo l'ape Regina, può visionare le notizie nelle diverse parti del mondo, figura 8.10. In questo senso quindi, esso fornisce un'indicazione dello stato di salute della Terra e viene utilizzato dalla protagonista del corto per far aderire allo sciopero anche gli animali degli ecosistemi che lei visita.

Come elementi tangibili, i cinque cubi assumono diverse funzionalità, e tra i cinque ne prevale uno, il cosiddetto cubo Main, da cui dipendono i restanti quattro cubi. Complessivamente, tali dispositivi svolgono la funzione di segnalare quando si verificano determinate situazioni di pericolo, evidenziati dall'illuminazione delle

Figura 8.10: Frame del cortometraggio in cui si vede l'ape Regina che riceve un avviso di pericolo, segnalato con il colore rosso, e quindi legge la relativa notizia.

facce laterali mediante i led all'interno del dispositivo, e vengono utilizzati nelle interazioni previste durante l'esperienza complessiva, interazioni che si esplicitano mediante quattro giochi.

Per quanto concerne la struttura narrativa dell'esperienza, questa si articola in quattro punti; simbolicamente l'evoluzione del cortometraggio è caratterizzata dal passaggio graduale del controllo della situazione climatica dall'ecosistema delle api, e in generale dal regno animale, alla specie umana, rappresentata in questo caso dalle nuove generazioni e quindi dai bambini. Dal punto di vista metaforico tale passaggio è anche sottolineato dall'utilizzo dei cubi nelle diverse parti della narrazione, come rappresentato nello schema in figura 8.11:

1. La situazione iniziale è caratterizzata dai 5 cubi a disposizione dell'ape Regina, la quale ha la visione d'insieme dello stato di salute del pianeta Terra e dei diversi ecosistemi. In questa fase l'utente non ha alcun ruolo ed è limitato ad osservare quanto presentato.

2. Il primo punto di svolta si verifica con l'affidamento del cubo principale a Pinnie, la protagonista del cortometraggio. A partire da questo passaggio il cubo assume un ruolo simbolico all'interno della narrazione e diventa una guida per l'utente. Questa fase si declina in due diversi punti di interazione che saranno approfonditi successivamente.

3. Il passaggio conclusivo si verifica infine con la consegna dei cubi all'utente ed è rappresentativo dell'acquisizione del controllo della situazione da parte dell'utente, il quale non dispone più dell'aiuto offerto da Pinnie; ora è quest'ultimo che deve intervenire e agire attraverso i cubi. A tal fine è necessaria la collaborazione con gli altri utenti, in quanto solo attraverso valori e comportamenti condivisi si possono ottenere cambiamenti duraturi e di impatto.

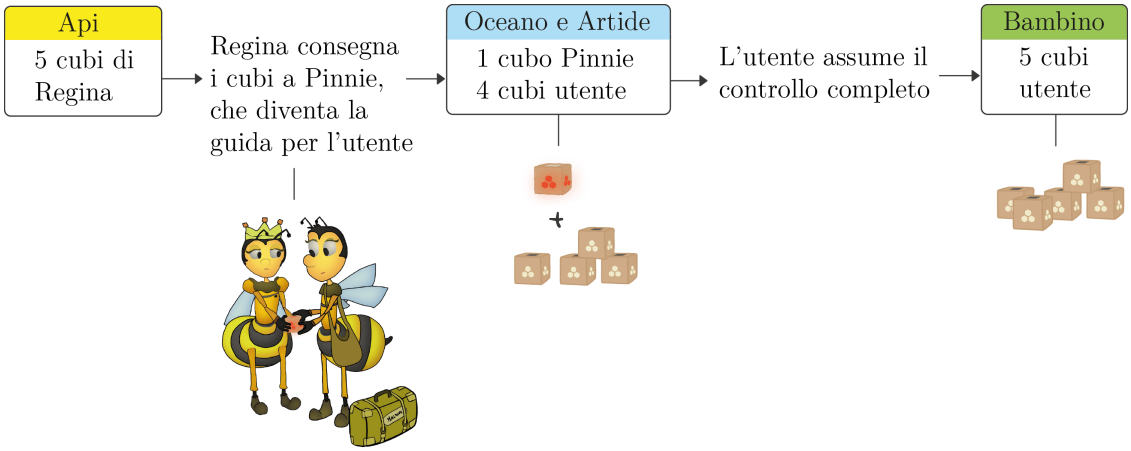


Figura 8.11: Evoluzione del ruolo dei cubi nell'esperienza

La struttura narrativa dell'installazione è costituita da quattro punti di interazione, o "punti di ingresso", nei quali è previsto l'intervento attivo dell'utente. Attraverso tali punti viene quindi richiesto all'utente di agire, e l'invito all'azione si esplicita in una serie di *call to action* che coinvolge i videogiochi proposti sui cubi fisici. Il percorso dell'installazione è stato predisposto in modo da aumentare gradualmente non solo l'immersione degli utenti nell'esperienza proposta ma anche le possibilità di utilizzo dei cubi e la complessità dei giochi.

L'inizio dell'esperienza avviene con lo scuotimento di uno dei cinque cubi posizionati sul tavolo; tale gesture avvia la riproduzione del cortometraggio. Il primo punto di interazione si sviluppa nella scena in cui si vede Regina usare il cubo per tenersi aggiornata sulle notizie principali riguardanti le condizioni in cui vivono alcune specie animali coinvolte nei cambiamenti climatici. Questa fase rappresenta di fatto il primo contatto strutturato dell'utente con i cubi, pertanto si è scelto di utilizzare una modalità di interazione ritenuta più semplice, e il gioco, *Unlock the cube*, prevede l'utilizzo contemporaneo di soli due cubi.
In questo punto si verifica il *trasporto* nel mondo fisico, a portata dell'utente, di una serie di contenuti narrativi del cortometraggio, virtuali. Si instaura quindi il primo collegamento tra due diversi spazi: l'irreale e il tangibile.

Il secondo punto d'ingresso si ha in seguito all'inizio del viaggio di Pinnie. A partire da questo istante il sistema dei cubi diventa pienamente operativo, consentendo

di arrivare gradualmente ad utilizzare contemporaneamente tutti e cinque i cubi. Questa seconda modalità di interazione, al contrario di quanto previsto per il gioco precedente e i due successivi, non si sviluppa più sul piano orizzontale e il concetto di verticalità assume un significato coerente con la tipologia di gioco descritta nel paragrafo 8.4.
Con il gioco *Ocean Getaway*, inizia per l'utente una fase esplorativa, in cui è necessario compiere delle scelte, che comportano la riuscita o l'insuccesso della missione proposta.

Attraverso il terzo punto di interazione viene invece proposto il gioco *Save Puppy*, che permette di sfruttare il pieno potenziale dei dispositivi. L'esito del gioco determina l'evoluzione del cortometraggio e nello specifico sono previsti due casi: nel caso di missione portata a termine con successo, il cortometraggio riprende e il viaggio di Pinnie prosegue; qualora invece, la missione fallisca, il cortometraggio riprende ma viene presentato uno scenario differente e quindi all'utente viene riproposta la missione. Pertanto il raggiungimento dell'obiettivo del gioco è una condizione necessaria per il proseguimento della narrazione e dell'esperienza.
Al fine di ridurre l'eventualità di ripetizione eccessiva del videogioco, che si trasformerebbe in frustrazione dell'utente e quindi interruzione dell'esperienza, sono stati adottati i seguenti accorgimenti per i tentativi successivi al primo:

- il tempo a disposizione per completare la missione verrà gradualmente incrementato;
- gli elementi del videogioco, la loro disposizione e le dinamiche verranno opportunamente modificate in modo randomico, introducendo una serie di variazioni nelle diverse ripetizioni del gioco.

Infine il punto di interazione che conclude l'esperienza proposta avviene con il passaggio del controllo nelle mani dell'utente. Così come avviene nel cortometraggio, il protagonista diventa ora un bambino, simbolo delle nuove generazioni, e questo ruolo viene impersonato dall'utente stesso. L'obiettivo del gioco conclusivo, *End the strike*, è quello di sbloccare l'alveare e quindi interrompere lo sciopero messo in atto dagli animali. A tal scopo, la scelta è ricaduta su una modalità di interazione che prevede un approccio partecipativo, in cui si richiede che il singolo utente collabori con altri utenti al fine di raggiungere l'obiettivo.

La struttura narrativa dell'installazione è illustrata in dettaglio nelle figure 8.12, 8.13 e 8.14.

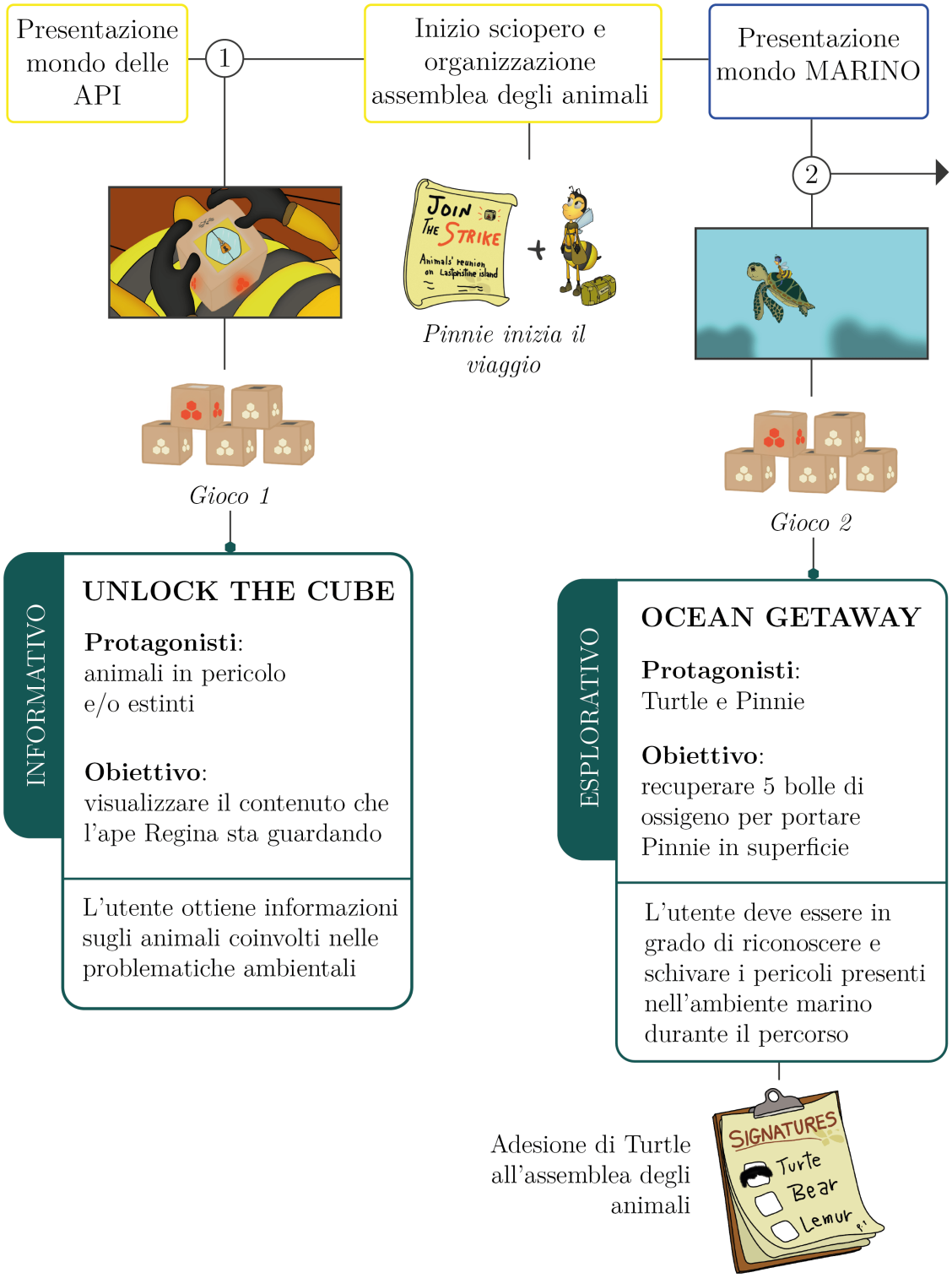


Figura 8.12: I parte della mappa narrativa dell'installazione [continua nella figura 8.13].

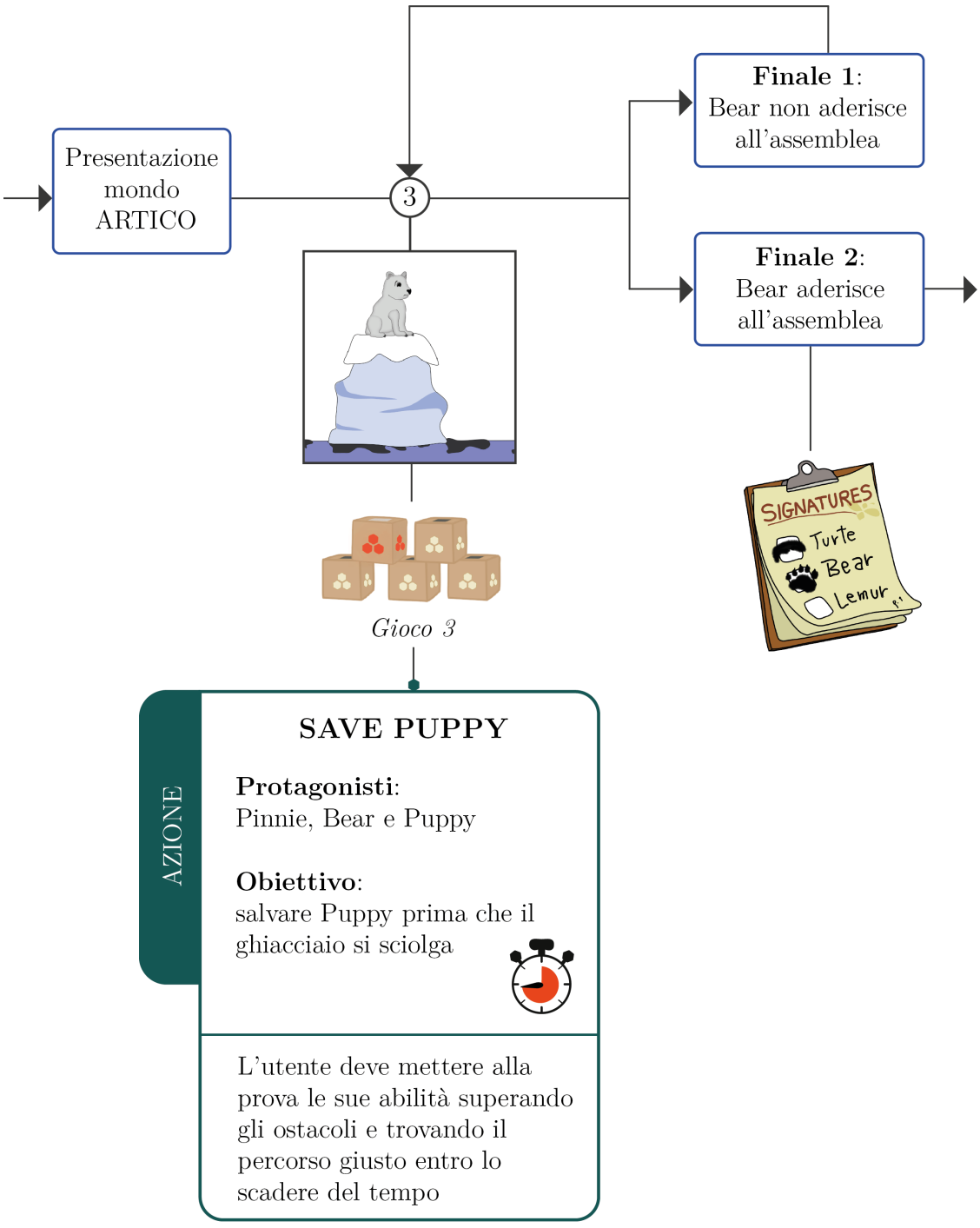


Figura 8.13: II parte della mappa narrativa dell'installazione [continua nella figura 8.14].

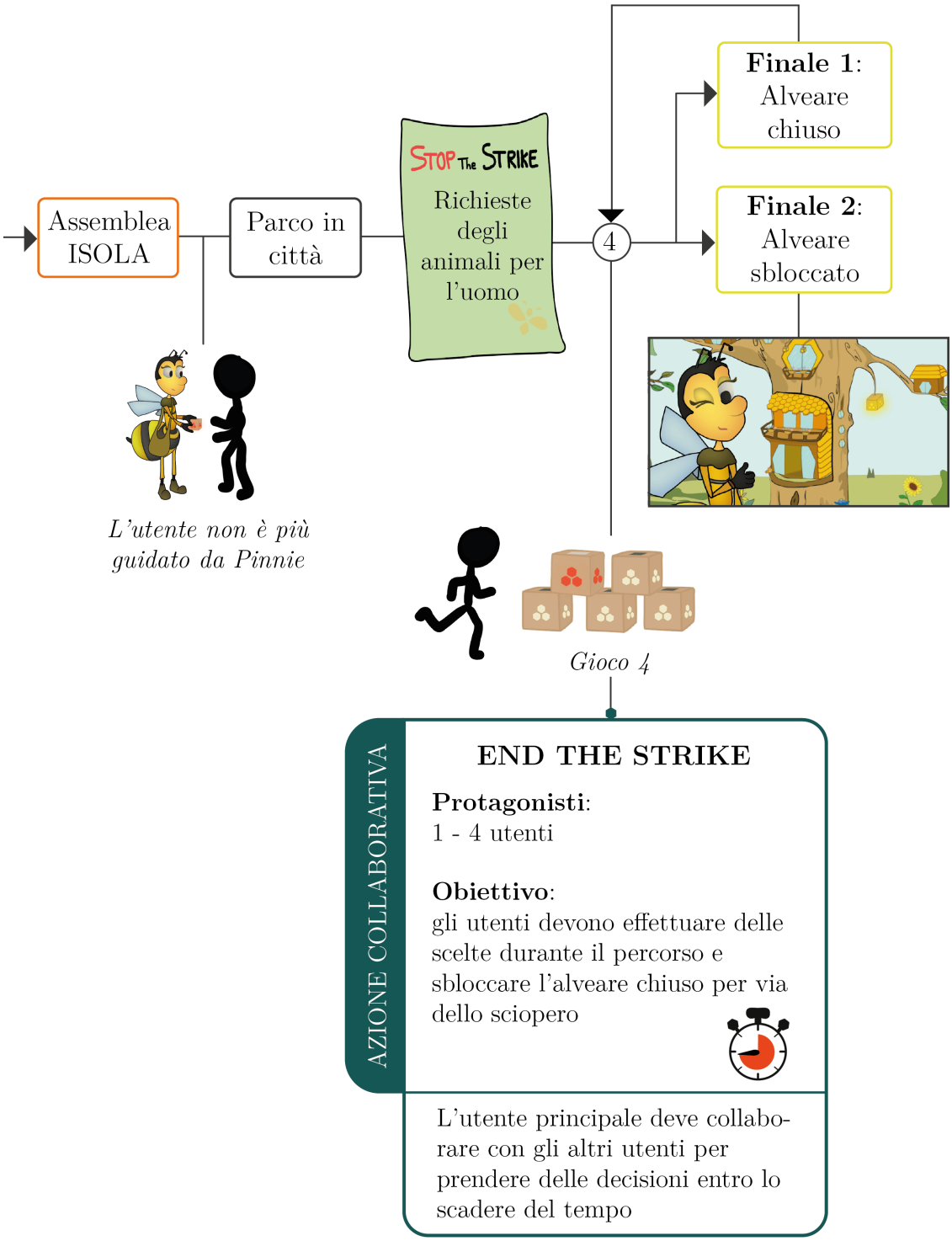


Figura 8.14: III parte della mappa narrativa dell'installazione.

Dal punto di vista informativo, nei quattro punti di interazione sono state sviluppate delle grafiche esplicative con lo scopo di illustrare all'utente le modalità di gioco e le azioni necessarie da compiere per poter manipolare i cubi ed entrare così nella fase di gioco vera e propria.
Nelle figure 8.15 e 8.16 sono riportati due esempi di tali grafiche nelle rispettive fasi di interazione del primo e del secondo gioco previsto dal sistema. Lo sfondo di ciascun gioco presenta un colore diverso che richiama l'ambiente in cui l'utente si trova nei rispettivi momenti della narrazione; per i primi due giochi vengono usati una tonalità del giallo per l'ambiente rappresentato dalle api e una tonalità del blu per indicare l'ambiente marino.

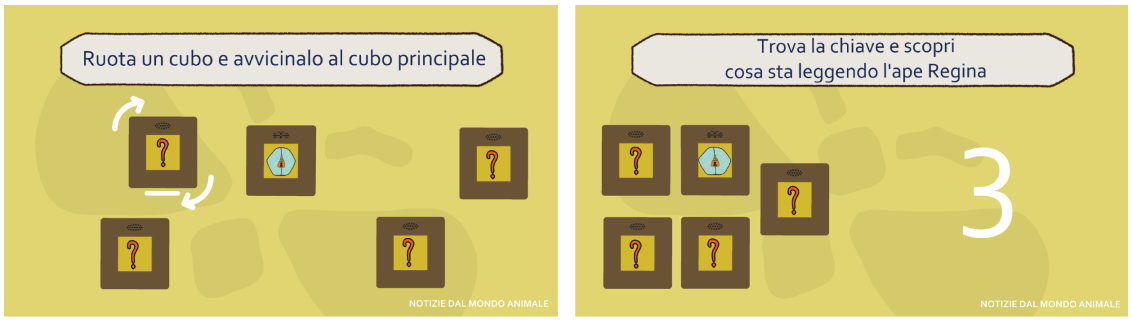


Figura 8.15: Grafica contenente le istruzioni e gli obiettivi che l'utente deve raggiungere per completare il gioco *Unlock the cube*. Il numero 3 nella figura indica il timer che prepara l'utente all'avvio del gioco.

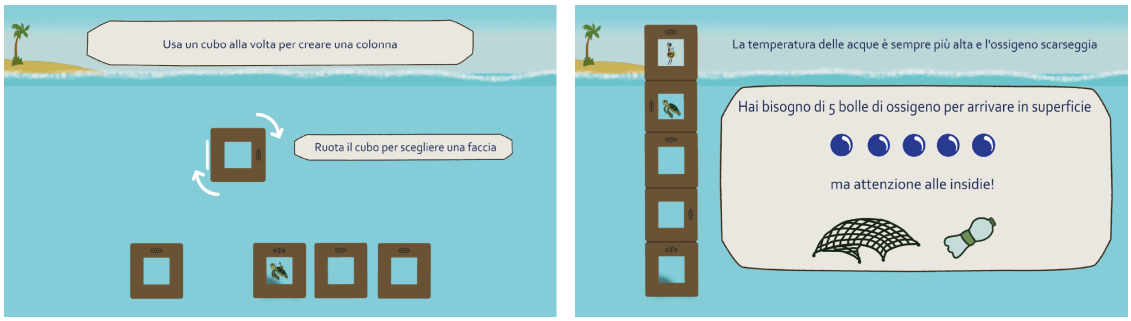


Figura 8.16: Grafica contenente le istruzioni e gli obiettivi che l'utente deve raggiungere per completare il gioco *Ocean Getaway*. Anche in questo caso è presente un timer.

8.4 Videogiochi

La scelta di focalizzare l'attenzione sul gaming come strumento base dell'interazione scaturisce dall'intenzione di voler creare un coinvolgimento di tipo ludico tra sistema e utente, efficace soprattutto per il tipo di target a cui è rivolto il progetto.

Come affermato dal game designer Eric Zimmerman in [10], "i giochi sono forse il primo sistema interattivo progettato che la nostra specie abbia inventato".
Con il progressivo aumento delle informazioni vi è stata la crescente necessità di rappresentare queste ultime tramite immagini. Inserendo tali immagini in esperienze di gioco, le informazioni assumono così un carattere interattivo, a differenza dei classici media lineari.

> "Media and culture in the Ludic Century is increasingly systemic, modular, customizable, and participatory. Games embody all of these characteristics in a very direct sense. [...] We live in a world of systems. [...] Games are machines of inputs and outputs that are inhabited, manipulated, and explored."[10]

Con l'espressione "secolo ludico" Zimmerman si riferisce al periodo storico del XXI secolo che prese il posto del precedente "secolo dell'informazione".

Nel libro *The gameful world* [11], Sebastian Deterding estende il concetto di gamification utilizzando il termine *gameful world*, e sulla base della distinzione teorizzata dal filosofo Roger Caillois, suggerisce di approcciarsi all'uso dei giochi e di andare oltre il piacere dell'intrattenimento lungo due dimensioni: la *paidia* e il *ludus*.

La prima dimensione si riferisce ai giochi senza regole, liberi ed improvvisati, mentre il ludus comprende le modalità di gioco formalizzate e basate su regole orientate per raggiungere degli obiettivi.
Si possono quindi distinguere quattro forme di gioco:

1. Serious games: *ludic wholes* o giochi veri e propri progettati e/o sviluppati per scopi non di intrattenimento.
2. Serious toys: *paidic wholes* o giochi progettati e/o sviluppati per scopi non di intrattenimento.
3. Playful design: *paidic elements* or qualities od oggetti non-giocattoli ed esperienze che usano elementi di design da giocattoli e/o vengono progettati per le esperienze di gioco.
4. Gamification (o gameful design): *ludic elements or qualities* od oggetti non-giochi ed esperienze che usano elementi di design da giochi e/o progettati per permettere le esperienze di gioco.

I giochi e il design del gioco influiscono enormemente sulla nostra cultura e dal momento che il loro scopo è creare contenuti interattivi piacevoli, diventano un mezzo di comunicazione fondamentale che si può applicare in qualsiasi contesto che prevede una progettazione caratterizzata da intrattenimento e divertimento.

Dal punto di vista della modalità del videogioco si può far riferimento al modello delle funzioni di comunicazione di Roman Jakobson, in cui la comunicazione tra

utente e dispositivo è "dominata dal contatto, o dalla funzione tattile, incentrandosi quindi sul canale fisico e sul lato stesso della connessione tra il mittente e il destinatario"[12].
Lev Manovich descrive ampiamente i nuovi media trattando numerosi aspetti di cui vengono di seguito riportati e argomentati in base alle caratteristiche del sistema CuBee progettato e sviluppato per la presente tesi.
A differenza dei vecchi media che erano costruiti in un determinato modo e a cui venivano associati funzioni fisse permanenti, "un nuovo oggetto mediale non è qualcosa che rimane identico a sé stesso all'infinito, ma è qualcosa che può essere declinato in versioni molto diverse tra loro. [...] I nuovi media sono caratterizzati dalla variabilità". Tale principio della variabilità, che Manovich associa ai media odierni, è strettamente legato all'aspetto della modularità, una caratteristica tipica della funzione che assumono i 5 cubi del sistema CuBee, ma che si limita alle combinazioni prestabilite in fase di progettazione in base agli obiettivi prefissati.
Tra i casi particolari del principio di variabilità che individua Manovich vi sono tre forme che possono essere utilizzate per caratterizzare l'insieme dei dispositivi di gioco sviluppati:

1. "Gli elementi costitutivi dei media vengono immagazzinati in un database; da questo provengono – preventivamente o a richiesta – un'enorme varietà di oggetti destinati all'utente finale, oggetti che variano per risoluzione, forma e contenuto. Tutto ciò non rappresenta solamente un'implementazione tecnologica del principio di variabilità, ma nell'era dei computer il database assurge a forma culturale a sé stante. Offre un modello preciso del mondo e dell'esperienza umana oltre ad influenzare il modo in cui l'utente elabora i dati che contiene".

2. "È possibile separare i livelli del «contenuto» (i dati) da quelli dell'interfaccia. Dagli stessi dati si possono creare interfacce diverse. Infatti, un nuovo oggetto mediale si può definire come una o più interfacce per l'accesso a un database multimediale".

3. "Un caso particolare della personalizzazione è l'*interattività ramificata* (chiamata anche «interattività su *menu*»). La definizione si riferisce a quei programmi in cui tutti i possibili oggetti che l'utente potrebbe visitare formano una struttura ad albero. Quando l'utente raggiunge un determinato oggetto, il programma gli propone una serie di scelte e gli consente di selezionare quella che preferisce. A seconda dell'opzione selezionata, l'utente procede lungo un determinato percorso. In questo caso le informazioni usate dal programma sono l'output del processo cognitivo dell'utente, anziché del proprio indirizzo elettronico o della posizione assunta dal proprio corpo".

Quest'ultimo punto è quello che caratterizza maggiormente i dispositivi di gioco

sviluppati in quanto ogni azione che viene proposta all'utente comporta delle successive azioni ramificate.
"Nel caso dell'interattività, l'utente gioca un ruolo attivo nel determinare l'ordine con cui accedere a degli elementi già generati in precedenza. È questo il tipo più semplice di interattività". Manovich individua tre tipologie di interazione in cui sia gli elementi sia la struttura dell'oggetto mediale vengono modificati o generati all'istante, in base all'interazione dell'utente con un certo programma. Si ha quindi l'*interattività aperta* che indica la tipologia di applicazioni variabili e che differisce dalla cosiddetta *interattività chiusa*, la quale utilizza elementi fissi organizzati in una struttura ramificata prestabilita, ed è in quest'ultima forma di interattività che possono essere inseriti i dispositivi di gioco. I cinque cubi come oggetti singoli "mantengono sempre la loro identità individuale" (in base al principio della modularità), si possono "interconnettere" in più di un oggetto.
I concetti che vengono espressi da Manovich riguardo ai nuovi media possono essere riassunti con la seguente descrizione:

> "I nuovi media sono interattivi. Diversamente dai vecchi media, in cui l'ordine di rappresentazione è fisso, oggi l'utente può interagire con un oggetto mediale. Grazie all'interazione l'utente può scegliere gli elementi da visualizzare o i percorsi da seguire, generando così un output personalizzato. In questo modo l'utente diventa anche coautore dell'opera."

I giochi rappresentano quindi uno strumento di innovazione che consente all'utente di sviluppare il senso di creatività e di pensiero, imparando nuovi modi di agire. Essi hanno il duplice scopo di approfondire parte delle tematiche illustrate nel cortometraggio e di facilitarne la comprensione, mettendo anche alla prova l'utente e verificando il recepimento di una serie di comportamenti positivi per l'ambiente e per l'uomo stesso.
Inoltre, affidando all'utente la possibilità di compiere determinate azioni per raggiungere degli obiettivi, stimola e motiva l'utente ad approcciarsi a questo tipo di interazione per poter scoprire gradualmente l'andamento della storia. Gli utenti dovranno quindi mettere in atto le proprie abilità, sia a livello individuale sia sotto forma di collaborazione con altri utenti, per poter raggiungere progressivamente gli obiettivi proposti, fino ad arrivare allo step conclusivo dell'esperienza che prevede la presa di coscienza delle situazioni illustrate e l'azione concreta per interrompere una serie di comportamenti dannosi.

Complessivamente il sistema CuBee prevede quattro punti di interazione da parte dell'utente durante la narrazione e la complessità delle abilità richieste aumenta gradualmente.
Sono stati sviluppati quattro giochi semplici dal punto di vista concettuale in modo che il messaggio trasmesso sia di facile comprensione per l'utente e si è deciso di soffermarsi maggiormente sulle particolari funzionalità e caratteristiche che contraddistinguono i dispositivi sviluppati:

- modularità: si parte da una situazione in cui l'utente prende consapevolezza del modo in cui devono essere adoperati i cubi, ossia della modularità che caratterizza i dispositivi; successivamente tale funzionalità dovrà essere sfruttata per creare percorsi più articolati. A seconda delle scelte e del modo in cui l'utente combina i dispositivi si creano percorsi diversi che danno origine a contenuti differenti.

- Display touchscreen: durante l'utilizzo dei cubi il display risulta il principale elemento, oltre alla modularità, che permette l'interazione tra il sistema e l'utente, in quanto consente di visualizzare un immediato riscontro successivo alle azioni dell'utente. Manovich definisce lo schermo come uno strumento che permette all'interazione di diventare reale. A differenza dello schermo classico, lo schermo odierno può essere definito come "schermo dinamico" che ha la funzione di mostrare contenuti che cambiano nel tempo. Tale tipologia di schermo ha la capacità di creare un'interazione tra contenuto e utente che l'autore indica come *regime di visione*, il quale "è reso possibile dal fatto che quella particolare immagine, un dipinto, un film o un programma televisivo riempie completamente lo schermo. Ecco perché ci infastidisce quando al cinema ci accorgiamo che l'immagine proiettata non coincide precisamente con i confini dello schermo: fa cadere l'illusione e ci dà la consapevolezza di ciò che esiste al di fuori dell'ambito rappresentativo. [...] Lo schermo ci presenta un mondo illusionistico virtuale e ci impone periodicamente delle azioni mediate".
  Inoltre l'integrazione del touchscreen consente un maggior coinvolgimento da parte dell'utente in quanto aumenta i metodi decisionali delle azioni, oltre alla combinazione dei dispositivi.

- Contenuti: i contenuti sviluppati nei videogiochi dei cubi riprendono gli elementi narrativi presenti nella storia illustrata tramite il cortometraggio. Si crea così una sorta di continuazione narrativa che cambia spazialmente mutando la propria forma e si adatta alle caratteristiche dei diversi media, dallo schermo del proiettore allo schermo dei cubi: "l'oscillazione tra segmenti illusionistici e interattivi della narrazione costringe l'utente a passare da uno stato mentale all'altro, a esercitare diverse forme di attività cognitiva".

- Grado di difficoltà: i giochi sono stati progettati in modo da consentire all'utente una graduale acquisizione di conoscenze e competenze. Si parte quindi da una situazione iniziale rappresentata dal primo gioco in cui l'utente ha un primo approccio con la modalità di funzionamento dei cubi, ossia l'accostamento tra di essi; fino ad arrivare all'ultimo step che richiede uno sforzo maggiore. Pertanto, seguendo un percorso si parte dal conoscere l'argomento per poi arrivare a delle sfide che mettono alla prova le abilità degli utenti e allo stesso tempo assimilano informazioni sul tema trattato.

- Modalità individuale e collaborativa: la possibilità di usare i dispositivi di gioco individualmente consente a ciascun utente di basarsi esclusivamente sulle proprie capacità nel decidere quali azioni compiere. D'altra parte la modalità collaborativa ha l'intenzione di evidenziare come le scelte compiute in base ad opinioni diverse sono uno strumento fondamentale per poter risolvere determinati problemi in modo più efficace e ricco, favorendo lo spirito di gruppo.

Di seguito vengono descritti ed illustrati in dettaglio i videogiochi progettati per il sistema CuBee.

8.4.1 Unlock the cube

L'obiettivo del gioco *Unlock the cube* è quello di visionare una serie di contenuti di cui l'ape Regina stava usufruendo nel cortometraggio e che non vengono mostrati all'utente. Quest'ultimo dovrà quindi sbloccare i 4 diversi contenuti racchiusi nei cubi e per fare ciò dovrà prima trovarne la chiave giusta. Il cubo principale, sul quale è rappresentato un lucchetto ha lo scopo di permettere l'accesso ai contenuti degli altri cubi, previa un'opportuna chiave, e il contenuto sbloccato verrà *distribuito* sui display di entrambi i cubi coinvolti nella connessione.

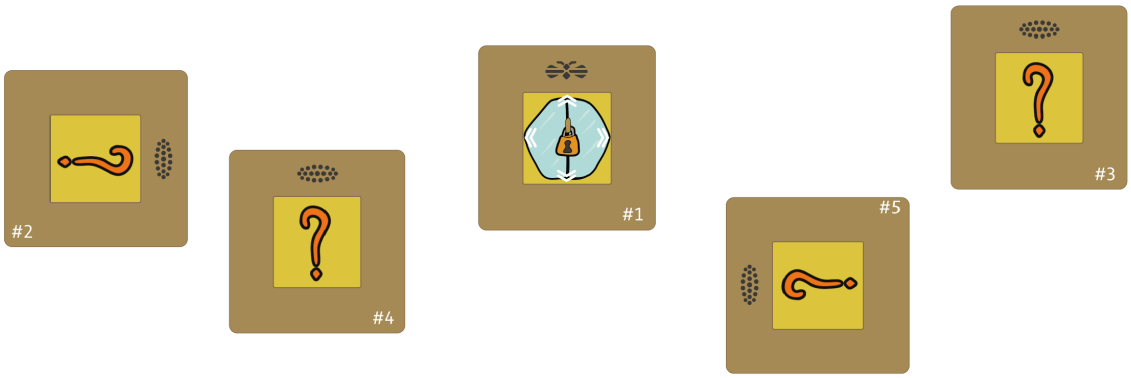


Figura 8.17: Stato iniziale dei cubi nel gioco *Unlock the cube*: il cubo principale, al centro, mostra una porta chiusa da un lucchetto e le quattro frecce lampeggianti sui lati invitano a collegare uno dei cubi secondari.

Ogni cubo racchiude un contenuto audiovisivo sotto forma di notizia, della durata di circa 15 secondi, riguardo ad uno dei seguenti argomenti:

- riscaldamento climatico e relativi effetti sull'uomo;
- disboscamento delle foreste tropicali e relativi effetti su specie animali come i lemuri e gli oranghi;

- inquinamento e relativi effetti sugli oceani e sugli organismi acquatici;
- scioglimento dei ghiacciai e relativi effetti su specie animali come gli orsi polari.

Il gioco prevede l'utilizzo contemporaneo di due soli cubi, uno dei quali è il cubo principale. Ciascun cubo secondario dispone di una sola chiave d'accesso per un determinato contenuto e tale chiave si trova su una delle quattro facce laterali del cubo. In figura 8.18 è mostrato il contenuto delle diverse facce del cubo 2; in questo caso la chiave corretta per sbloccare il rispettivo contenuto si trova sulla faccia C. Nei restanti tre cubi, la disposizione dei contenuti è diversa, sfruttando tutte le combinazioni possibili.

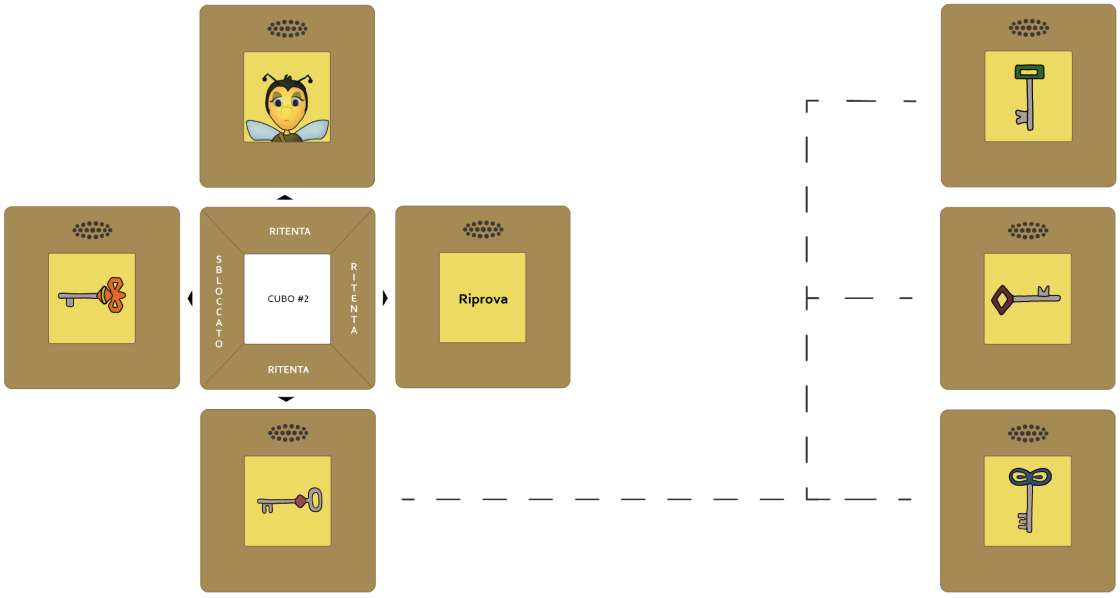


Figura 8.18: Rappresentazione delle facce laterali del cubo 2 e delle grafiche mostrate sul display quando tali facce sono coinvolte in una connessione. Sulla destra sono illustrate le restanti chiavi sbagliate previste per gli altri tre cubi .

Per poter accedere al contenuto del cubo 2, l'utente dovrà quindi trovare la chiave corretta affiancando quest'ultimo al cubo principale su una delle sue facce; il lucchetto verrà sbloccato solo quando il cubo 2 sarà connesso al cubo principale attraverso la faccia C, contenente la chiave corretta, come mostrato in figura 8.19a.

In seguito al ritrovamento della chiave corretta ed allo sblocco del lucchetto, sarà possibile visionare il contenuto associato, disposto opportunamente sui display dei due cubi coinvolti nella connessione, come mostrato in figura 8.19b. Alla fine del contenuto il lucchetto verrà nuovamente chiuso mentre il cubo secondario verrà disattivato. In seguito allo sblocco di tutti e quattro i cubi, sul display del cubo principale comparirà un pulsante che l'utente deve premere per riprendere la visione

del cortometraggio; qualora tale pulsante non venisse premuto entro 7 secondi, il sistema continuerà automaticamente la riproduzione.

(a) **(b)**

Figura 8.19: In (a) affiancamento del cubo 2 al cubo principale, utilizzando la faccia corretta. In (b) un esempio del contenuto mostrato sui display dei due cubi coinvolti nella connessione.

8.4.2 Ocean Getaway

Ocean Getaway è invece il secondo gioco vero e proprio in cui l'utente si trova a dover affrontare la prima missione proposta durante l'esperienza.
L'obiettivo del gioco consiste nel salvataggio della protagonista Pinnie, che si trova sul fondo dell'oceano e sta esaurendo l'ossigeno a disposizione. Per completare la missione, l'utente dovrà comporre una colonna sovrapponendo i cubi a disposizione con i display rivolti verso di sé e durante il percorso dovrà raccogliere cinque bolle di ossigeno entro l'ultimo cubo posizionato in cima alla colonna, facendo attenzione agli ostacoli previsti, come le reti da pesca e le bottiglie di plastica. Tali pericoli infatti, se non evitati dall'utente, comporteranno la perdita delle bolle di ossigeno raccolte e pertanto sarà più difficile completare il gioco.
La verticalità data dai cubi sovrapposti per formare una colonna indica il movimento naturale di risalita e simbolicamente rappresenta una sorta di termometro che indica la temperatura, sempre più alta, man mano che ci si avvicina alla superficie. D'altra parte le bolle di ossigeno da raccogliere evidenziano ulteriormente la scarsità di ossigeno degli ambienti marini e che gli organismi marini devono fronteggiare. Pertanto, ogni bolla raccolta indica una quantità di ossigeno recuperata.

Nello stato iniziale del gioco rappresentato in figura 8.20, il cubo 1, rappresentante il cubo Main, è l'unico dispositivo con il display acceso che mostra Turtle e Pinnie, i due personaggi protagonisti del gioco. Inoltre sono visibili le bolle di ossigeno raccolte dall'utente durante la risalita. L'utente inizia da una situazione con 2 bolle di ossigeno. I restanti quattro cubi sono spenti e vengono attivati solamente quando vengono posti al di sopra del cubo principale.

Ciascuno dei quattro cubi può essere ruotato per scegliere una faccia da posizionare. In figura 8.21 è mostrato il contenuto di ciascuna faccia di un cubo; in questo caso viene preso in considerazione il cubo 2, mentre per i restanti tre cubi i contenuti

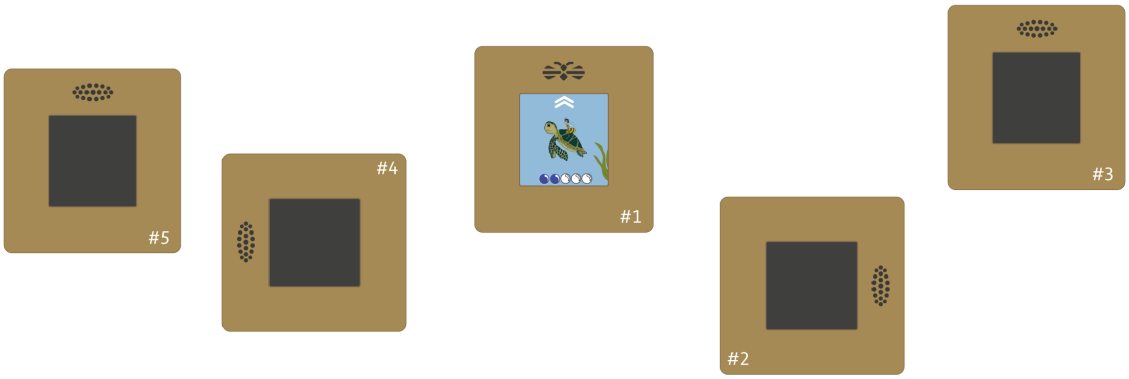


Figura 8.20: Vista dall'alto dello stato iniziale del gioco *Ocean Getaway*.

sono gli stessi ma disposti sfruttando tutte le combinazioni possibili delle quattro facce.

Figura 8.21: Contenuti delle quattro facce del cubo 2 nel gioco *Ocean Getaway*.

Nello specifico, a ciascuna faccia sono associati i seguenti elementi:

- BONUS: sul display compare una bolla di ossigeno che l'utente può raccogliere toccando lo schermo.
- OSTACOLO: sul display compaiono alcune bolle di ossigeno che si muovono velocemente in orizzontale e tra di esse vi è una bottiglia di plastica che si

muove anch'essa in orizzontale ma lentamente. Quest'ultima rappresenta l'ostacolo, in quanto se toccata dall'utente causa il decurtamento di una bolla di ossigeno.

- DANNO: sul display compare una rete da pesca. In questo caso l'utente non ha alcun controllo e perde una bolla di ossigeno; inoltre i due personaggi scendono di una posizione, ritrovandosi sul display del cubo sottostante.
- STABILE: i personaggi risalgono senza alcun pericolo.

In seguito al posizionamento di tutti e quattro i cubi sopra il cubo Main, si possono verificare due possibili situazioni, visibili in figura 8.22.

(a) **(b)**

Figura 8.22: In (a) rappresentazione del finale in caso di obiettivo raggiunto, mentre nel cubo di destra, (b), è visibile il caso in cui l'obiettivo non è stato raggiunto.

Nello specifico, nel caso in cui l'utente abbia raccolto tutte e cinque le bolle di ossigeno (le icone delle cinque bolle sono colorate tutte di blu) entro il posizionamento del quinto cubo nella colonna, sul display dell'ultimo cubo compare Pinnie davanti ad un'isola indicando che la missione è stata completata con successo.
Inoltre compare un pulsante, *Continua storia*, necessario per riprendere la proiezione del cortometraggio e quindi proseguire con la narrazione.

In caso contrario, qualora l'utente non raccogliesse tutte e cinque le bolle (le icone delle bolle sono tutte colorate di rosso), sul display rimangono i due protagonisti e compare il pulsante *Riprova* che invita l'utente a riprovare a completare la missione.

Per quanto riguarda l'illuminazione delle facce dei cubi, queste assumono un colore diverso a seconda del numero di bolle di ossigeno raccolte dall'utente:

- 1 bolla: il colore utilizzato è il rosso. Rappresenta la situazione in cui Turtle e Pinnie sono al limite della sopravvivenza.
- 2 bolle: il colore utilizzato è il giallo.
- 3 o 4 bolle: il colore utilizzo è il blu.
- 5 bolle: il colore utilizzato è il verde.

8.4.3 Save Puppy

Save Puppy è il terzo gioco previsto dal sistema CuBee. In questo caso l'utente si trova a dover affrontare una situazione che prevede un grado di complessità maggiore, in quanto si tratta di un gioco a tempo. È infatti presente un timer rappresentato, oltre che da un'icona verticale i cui colori cambiano a seconda della situazione, anche dal ghiacciaio su cui si trova Puppy che scompare gradualmente rappresentando lo scioglimento e l'aumento crescente del riscaldamento climatico. L'obiettivo, infatti, è trovare il percorso più breve per raggiungere il cucciolo d'orso polare e salvarlo entro lo scadere del tempo.

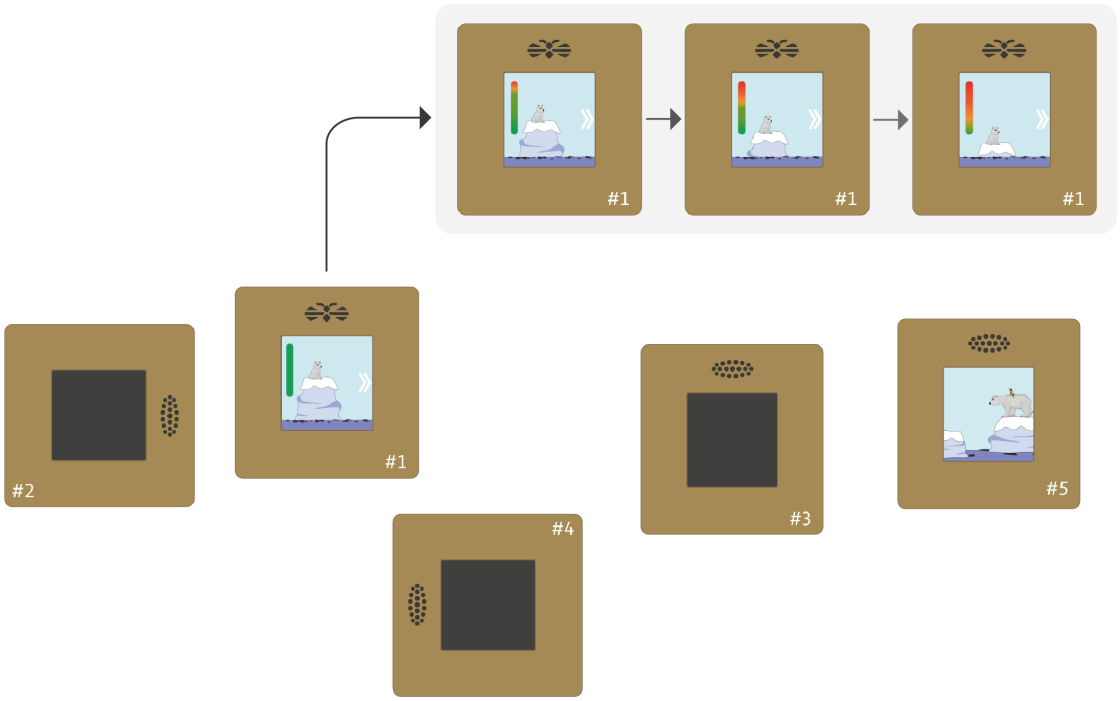


Figura 8.23: Situazione iniziale del gioco *Save Puppy*: il cubo 1 mostra due frecce lampeggianti sul bordo destro del display per indicare all'utente la faccia da usare per accostare un cubo nel collegamento finale. In alto a destra viene mostrata l'evoluzione della situazione di Puppy, con la diminuzione del tempo utile per essere salvato e il graduale scioglimento del ghiacciaio.

La situazione iniziale, mostrata in figura 8.23, prevede solamente i cubi 1 e 5 accesi, mentre i restanti tre dispositivi sono spenti e sono gli unici utilizzabili dall'utente. Il cubo 1 mostra Puppy bloccato su un ghiacciaio che si sta lentamente sciogliendo. Nel cubo 5 sono invece presenti i due protagonisti del gioco, Bear e Pinnie, i quali devono muoversi attraverso i restanti cubi fino a raggiungere il cubo con Puppy. In questo caso gli orsi non possono toccare l'acqua e quindi nuotare, in quanto è contaminata da enormi chiazze di petrolio il cui contatto con la pelliccia degli orsi

potrebbe provocare loro gravi danni, nonché la morte. Pertanto, nel gioco l'acqua è considerata come il maggior pericolo per i due protagonisti.

Ciascuno dei restanti tre cubi utilizzabili dall'utente per comporre il percorso contiene diversi contenuti e scene. Tali contenuti sono illustrati nelle figure 8.24, 8.25 e 8.26.

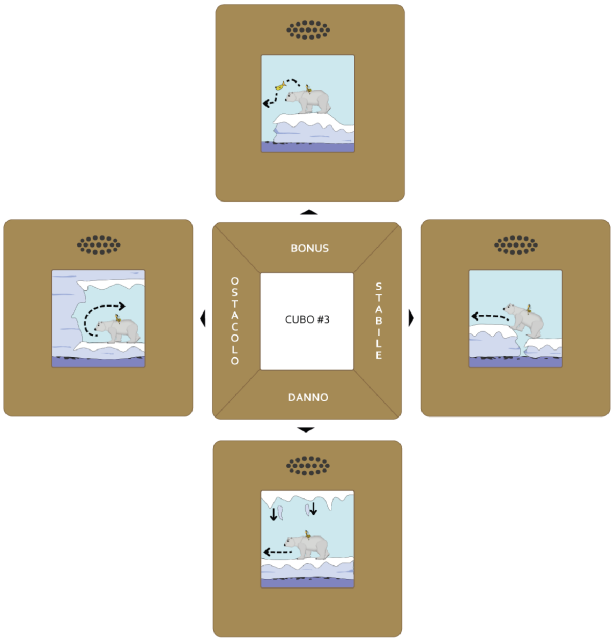


Figura 8.24: Distribuzione degli elementi Ostacolo, Bonus, Danno e Stabile sulle facce del terzo cubo.

Le facce del terzo cubo prevedono i seguenti contenuti, illustrati in figura 8.24:

1. STABILE: *ghiaccio stabile*, il percorso non presenta ostacoli, Bear e Pinnie possono proseguire normalmente.
2. BONUS: *pesce bonus*, toccando il pesce il tempo a disposizione viene congelato, guadagnando 5 secondi in più.
3. OSTACOLO: *passaggio bloccato*, Bear e Pinnie ritornano nel cubo precedentemente collegato.
4. DANNO: *ghiaccio si stacca*, durante il percorso si staccheranno dei pezzi di ghiaccio appuntiti dall'alto. Bear e Pinnie devono schivarli per non perdere la vita, nonché finire il gioco. L'utente dovrà quindi essere abile a toccare in fretta i pezzi di ghiaccio per disintegrarli prima che cadano sui due protagonisti.

Figura 8.25: Distribuzione degli elementi Ostacolo, Bonus, Danno e Stabile sulle facce del secondo cubo.

Le facce del secondo cubo prevedono i seguenti contenuti, illustrati in figura 8.25:

1. BONUS: *pesce bonus*, toccando il pesce il tempo a disposizione viene congelato, guadagnando 5 secondi in più. Per poter proseguire il percorso occorre collegare un altro cubo sulla faccia superiore di quest'ultimo.
2. OSTACOLO: *ghiaccio scorrevole*, appena l'utente collega questa faccia con un cubo, Bear e Pinnie rimangono nel cubo collegato in partenza. Il blocco di ghiaccio di questa faccia scorre verso destra e sinistra e bisogna toccare al momento giusto, un qualsiasi punto del display, per far saltare i due personaggi dal cubo in cui si trovano al blocco di ghiaccio che scorre nel cubo vicino, evitando la caduta in acqua.
3. DANNO: *ghiaccio instabile*, appena i due si spostano sul blocco di ghiaccio mostrato, quest'ultimo si scioglie e i due protagonisti cadono in acqua. Bear e Pinnie ricompaiono sul display del cubo precedentemente collegato.
4. STABILE: il percorso non presenta ostacoli, Bear e Pinnie possono proseguire normalmente.

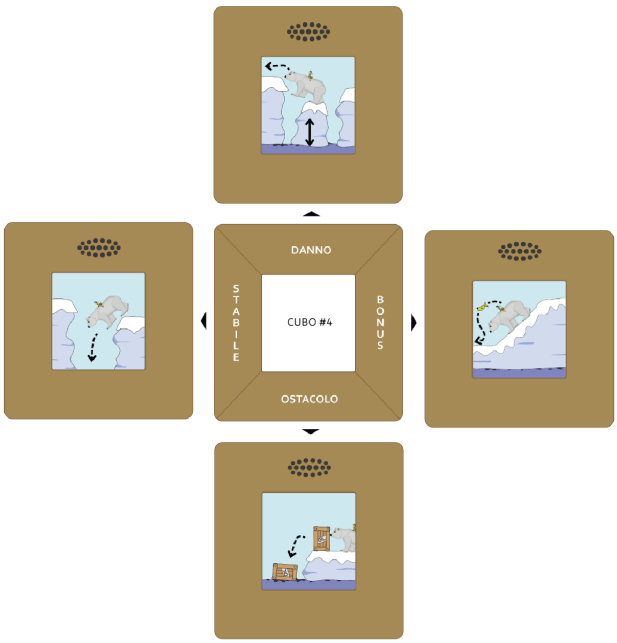
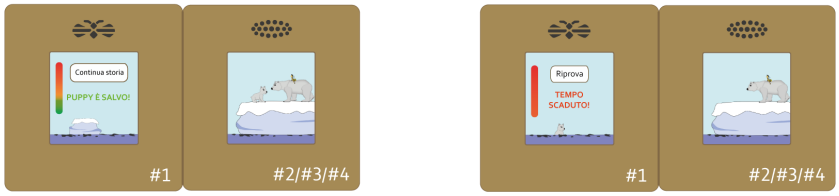


Figura 8.26: Distribuzione degli elementi Ostacolo, Bonus, Danno e Stabile sulle facce del quarto cubo.

Le facce del quarto cubo prevedono i seguenti contenuti, illustrati in figura 8.26:

1. BONUS: *pesce bonus*, toccando il pesce il tempo a disposizione viene congelato, guadagnando 5 secondi in più. Per poter proseguire il percorso occorre collegare un altro cubo sulla faccia inferiore di quest'ultimo.
2. DANNO: *ghiaccio scorrevole*, Bear e Pinnie rimangono fermi sul blocco di ghiaccio sulla destra mentre il ghiacciaio centrale scorre in verticale e in orizzontale. L'utente dovrà cliccare sul punta in prossimità del ghiacciaio più a sinistra appena il ghiacciaio centrale si trova alla massima altezza, altrimenti Bear e Pinnie cadranno, perdendo la vita e quindi l'utente dovrà ricominciare il gioco.
3. STABILE: il percorso non presenta ostacoli, Bear e Pinnie possono proseguire normalmente. Il cubo successivo deve essere collegato sulla faccia inferiore per poter proseguire il percorso.
4. OSTACOLO: *cassa di legno*, l'utente deve far scorrere Bear e Pinnie in orizzontale fino alla cassa di legno per poterla buttare in acqua e usarla come base di appoggio e proseguire il percorso.

Quando l'utente arriva ad effettuare il collegamento finale, del cubo in cui è presente Puppy con gli altri quattro cubi, si possono verificare due situazioni, come mostrato in figura 8.27. Qualora la missione venga completata, sul display del cubo Main compare il pulsante *Continua storia* che consente di proseguire la proiezione del cortometraggio. In caso contrario, ossia l'utente non ha raggiunto l'obiettivo in tempo, compare il pulsante *Riprova* che invita l'utente a ricominciare il gioco.



(a) **(b)**

Figura 8.27: I due cubi in (a) mostrano il caso in cui l'utente è riuscito a compiere la missione salvando Puppy, mentre a destra, (b), sono riportate le grafiche mostrate sui display in caso di perdita del gioco.

Infine, per quanto riguarda l'illuminazione delle facce dei cubi, durante il gioco sono visibili i seguenti colori:

- bianco, durante il percorso stabile.
- Rosso, nei casi in cui il tempo scada prima di arrivare all'ultimo collegamento con il cubo Main o quando si presentano danni e ostacoli che non vengono superati.
- Giallo, quando l'utente raccoglie i bonus e supera gli ostacoli.
- Verde, nel caso di vincita del gioco.

8.4.4 End the Strike

End the Strike rappresenta il gioco conclusivo dell'esperienza proposta e prevede un grado di complessità maggiore rispetto ai giochi precedenti. L'utente infatti non dovrà agire individualmente ma avrà bisogno della collaborazione di almeno un altro utente, fino ad un totale di 4 giocatori, per riuscire a portare a termine la missione, ossia porre fine allo sciopero degli animali e sbloccare l'alveare delle api. L'ambientazione è quella di una città che vede come protagonista un bambino in cui gli utenti si possono immedesimare. L'obiettivo è ripulire la città dai rifiuti, compiendo una serie di scelte. In particolare, gli utenti dovranno combinare i cubi a disposizione per creare un percorso durante il quale verranno proposte loro delle scelte che hanno un impatto sul punteggio finale, ossia l'indice di efficienza, fondamentale per poter accedere all'ultimo step prima di completare il gioco. Anche

in questo caso si tratta di un gioco a tempo.
In figura 8.28 è illustrato il caso in cui tutte le combinazioni dei cubi sono state effettuate in modo da ricreare il percorso corretto.

Nello specifico, il cubo Main, 1, rimane fisso e rappresenta il punto di arrivo che segna la fine del percorso. Sul display di tale cubo compaiono sovrapposti i rifiuti che l'utente dovrà raccogliere nei restanti cubi. L'avvio del gioco è visibile sul cubo 5 che mostra la figura del personaggio il quale dovrà essere guidato dagli utenti per attraversare il percorso creato. Nella fase iniziale sono quindi accesi solamente il cubo 1 e il cubo 5, mentre i restanti cubi sono spenti.

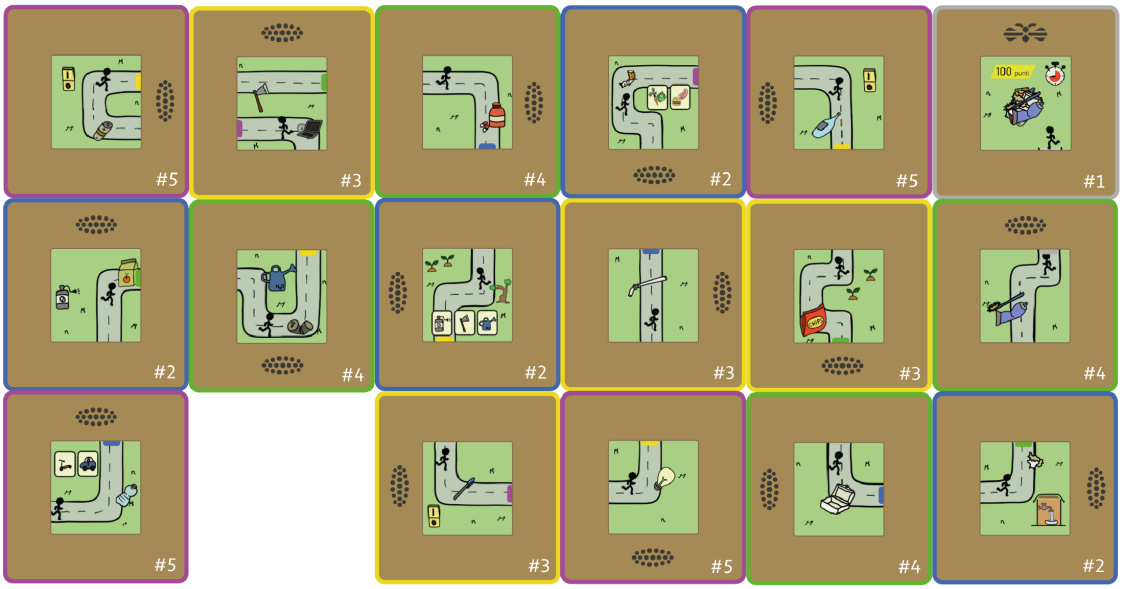


Figura 8.28: Percorso corretto del gioco *End the Strike*. Il cubo principale, in alto a destra, rimane fisso, mentre i restanti cubi devono essere collegati partendo dal 5, in basso a sinistra, su cui è visibile l'inizio del gioco. Il colore dei contorni rappresenta l'illuminazione delle facce laterali dei cubi.

Ciascun cubo presenta un colore diverso, ottenuto tramite l'illuminazione dei led, in modo da identificare ogni utente, e quindi si ha il colore viola per il cubo 5, verde per il cubo 4, giallo per il cubo 3 e blu per il cubo 2. Tali colori vengono utilizzati anche sui display per guidare le azioni degli utenti. Infatti su ciascun cubo è visibile alla fine della strada illustrata un'indicazione con il colore del successivo cubo da collegare, nonché il turno di ciascun giocatore; in questo modo ogni utente avrà lo stesso numero di interventi e si limiteranno i conflitti decisionali tra di essi. Una volta chiamato l'utente che dovrà intervenire, egli dovrà ruotare il proprio dispositivo di gioco e scegliere quale faccia collegare al cubo precedente; solamente nel caso in cui il collegamento sia corretto, il personaggio sul display comparirà nel cubo appena collegato, in caso contrario l'utente dovrà effettuare una nuova scelta.

Come nei precedenti giochi, ogni cubo contiene sulle proprie quattro facce dei contenuti diversi, in questo caso parti che compongono un'unica strada, che di seguito vengono descritti in base al numero identificativo del dispositivo.

Cubo 1

Il cubo Main rimane fisso e sul display cambia il contenuto a seconda delle fasi di gioco. Durante la creazione del percorso, sul display si accumulano sovrapposti i diversi rifiuti che gli utenti dovranno raccogliere su ciascun cubo. Un timer mostra lo scorrere del tempo mentre il numero sulla sinistra indica il punteggio di efficienza raggiunto dai giocatori e varia a seconda delle scelte effettuate fino ad un massimo di 100 punti.

Cubo 5

- Faccia A: contiene il personaggio nella posa iniziale. Sono inoltre presenti due icone cliccabili, una rappresentante un monopattino e la seconda contenente un'auto. Un primo tocco su queste icone farà comparire dei pop-up con brevi informazioni e dati su ciascun mezzo; cliccando una seconda volta l'utente effettuerà la scelta tra i due oggetti. A seconda della scelta effettuata, il personaggio proseguirà il percorso con il rispettivo mezzo di trasporto: nel caso dell'auto il punteggio di efficienza finale non verrà incrementato, mentre la scelta del monopattino contribuirà al punteggio finale con 20 punti.
  Questa faccia richiede la raccolta della seguente tipologia di rifiuto: bottiglietta di plastica.
- Faccia B: è presente un interruttore della luce, il quale, se cliccato darà la possibilità agli utenti di visionare per 5 secondi la piantina generale dell'intero percorso sul cubo Main. L'interruttore della luce rappresenta simbolicamente lo spreco energetico, pertanto la sua accensione comporterà una riduzione dell'efficienza di 10 punti. Tale elemento è presente anche in due ulteriori punti quindi occorre memoria e collaborazione per ricreare il percorso senza incidere negativamente sul punteggio finale.
  Questa faccia richiede la raccolta della seguente tipologia di rifiuto: pila usurata.
- Faccia C: contiene una lampadina a incandescenza fulminata, che l'utente deve raccogliere.
- Faccia D: contiene un interruttore della luce.
  Questa faccia richiede la raccolta della seguente tipologia di rifiuto: bottiglia di vetro.

Cubo 4

- Faccia A: il rifiuto da raccogliere è una lattina usata. Inoltre occorre raccogliere anche l'annaffiatoio con l'acqua che servirà successivamente nel gioco.
- Faccia B: i rifiuti da raccogliere sono un barattolo di plastica e delle pastiglie scadute sfuse.
- Faccia C: il rifiuto da raccogliere è un contenitore di polistirolo.
- Faccia D: i rifiuti da raccogliere sono uno spazzolino usato e un tubetto del dentifricio.

Cubo 3

- Faccia A: i rifiuti da raccogliere sono un CD e un pc non più funzionante. Occorre inoltre raccogliere un'ascia che servirà successivamente nel gioco. Un primo tocco sull'icona farà comparire un pop-up con brevi informazioni e dati riguardo il disboscamento; il secondo tocco permetterà di effettuare la raccolta dell'elemento.
- Faccia B: contiene un interruttore della luce. Il rifiuto da raccogliere è una penna consumata.
- Faccia C: il rifiuto da raccogliere è una cannuccia di plastica.
- Faccia D: il rifiuto da raccogliere è una busta di patatine.

Cubo 2

- Faccia A: contiene l'icona di uno spruzzatore per pesticidi. Quando viene cliccato per la prima volta compare un pop-up con una breve spiegazione dei danni che produce; il secondo tocco permetterà la raccolta dell'elemento. Occorre raccogliere questo elemento in quanto servirà successivamente nel gioco.
  Il rifiuto da raccogliere è un contenitore di succo di frutta in Tetra Pak.
- Faccia B: sono visibili delle piantine nel prato e tre icone con gli elementi raccolti precedentemente, ossia lo spruzzatore di pesticidi, l'ascia e l'innaffiatoio con l'acqua. L'utente dovrà scegliere uno dei tre elementi; solamente l'acqua incrementerà il punteggio finale mentre le restanti azioni comporteranno un decremento dell'indice di efficienza.
- Faccia C: sono presenti due icone, la prima ritrae delle verdure mentre la seconda contiene della carne. Cliccando su ciascuna icona una prima volta compaiono dei pop-up con delle brevi spiegazioni e dati che questi alimenti hanno sull'ambiente, nonché sull'uomo; un secondo tocco consentirà di effettuare la scelta tra i due elementi. La scelta delle verdure incrementerà il punteggio finale, mentre la scelta della carne comporterà un decremento.
  Il rifiuto da raccogliere è un mozzicone di sigaretta.

- Faccia D: contiene un rubinetto con l'acqua che scorre. Se verrà cliccato una prima volta comparirà un pop-up contenente dati sulla quantità di acqua sprecata in ambito domestico, ad esempio durante il lavaggio dei denti. Il secondo tocco permetterà di chiudere il rubinetto e ciò comporterà un incremento del punteggio finale, mentre nel caso in cui questo elemento non venga considerato, e quindi non verrà cliccato una seconda volta, si avrà una diminuzione del valore di efficienza.
  Il rifiuto da raccogliere consiste in un mucchio di carta usata.

Per poter effettuare la seconda parte del gioco finale, visibile in figura 8.29, occorre arrivare nell'ultimo collegamento con un punteggio minimo di 60 punti, altrimenti il gioco dovrà essere ripetuto.

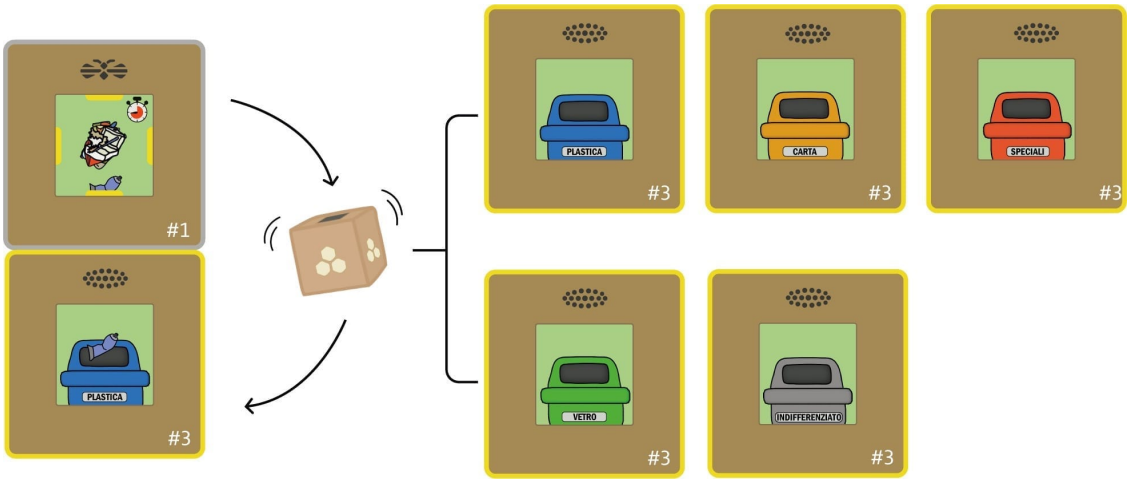


Figura 8.29: Step finale di *End the Strike*. In questo caso viene preso in considerazione il turno dell'utente che usa il cubo 3. I quattro cubi sulla destra rappresentano i contenuti delle quattro facce di ciascun cubo, uguali anche per i restanti tre cubi.

Nello specifico, sul cubo Main rimangono visibili i rifiuti raccolti e in alto a destra l'icona del timer con lo scorrere del tempo. Il cubo 1 rimane fisso e guida gli utenti richiamandoli uno ad uno a secondo del colore che compare sui bordi del display e sulle facce laterali.
L'obiettivo è fare la raccolta differenziata nel modo corretto ed entro il tempo a disposizione. Ciascun cubo contiene cinque contenitori per la differenziata: carta, plastica, vetro, indifferenziato e speciali. Attraverso lo shake del singolo cubo ciascun utente, confrontandosi anche con gli altri giocatori, potrà scegliere quale contenitore è più adatto per raccogliere il rifiuto che gli viene proposto sul cubo Main. Il rifiuto da utilizzare è sempre quello in primo piano sul mucchio raccolto che diminuisce man mano che vengono differenziati i rifiuti.
Se il gioco viene concluso correttamente, sullo schermo di proiezione del cortometraggio comparirà la protagonista Pinnie con l'alveare delle api sbloccato. In caso

contrario il gioco dovrà essere ripetuto.

Quest'ultimo gioco viene proposto agli utenti con l'obiettivo di stimolare la collaborazione e renderli consapevoli del fatto che per poter rimediare e migliorare la situazione climatica, ciascun individuo non può apportare un cambiamento radicale e immediato ma può agire iniziando dai piccoli gesti quotidiani, riflettendo e rivalutando le proprie abitudini in un'ottica più sostenibile e rispettosa dell'ambiente.

8.5 Implementazione software

L'interazione tra i dispositivi di gioco e il cortometraggio richiede uno strumento che elabori una serie di input provenienti dai cubi e traduca tali segnali in azioni sul cortometraggio. L'installazione si compone di quattro elementi, come da figura 8.30: (1) i cinque cubi, (2) un computer, (3) un impianto audio e (4) un proiettore video.

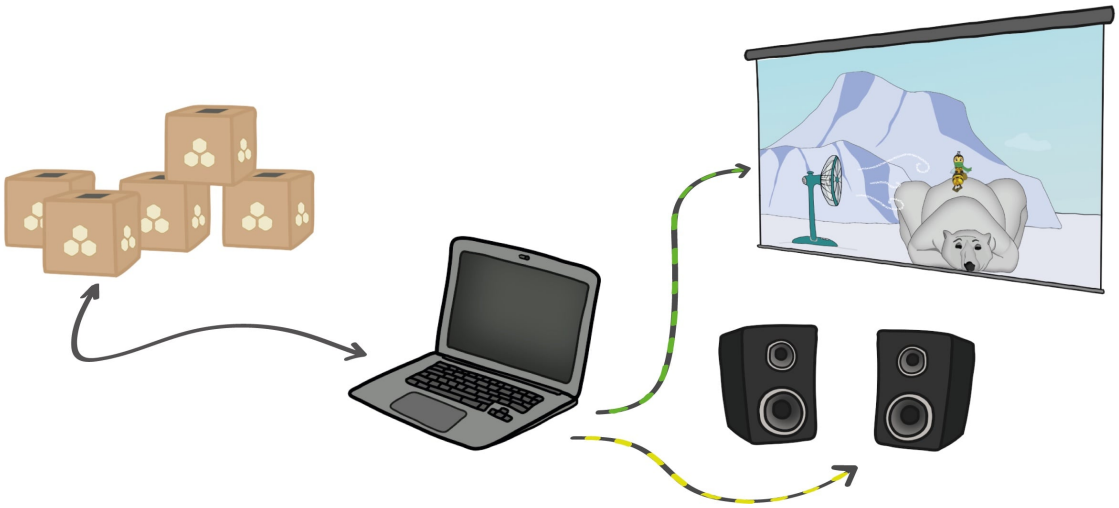


Figura 8.30: Rappresentazione degli elementi alla base dell'installazione: il ruolo chiave è assunto dal computer, il quale comunica con i cubi e fornisce il segnale video per il proiettore e il segnale audio per l'impianto del museo.

Il computer è l'elemento centrale dell'installazione ed esso gestisce non solo i segnali video diretti al proiettore, ma anche i segnali destinati all'impianto audio. A tal scopo si è quindi reso necessario uno strumento software che potesse svolgere i seguenti due compiti fondamentali: ricevere/inviare messaggi OSC e gestire la proiezione del cortometraggio. I principali software che rispondono a tale necessità

sono Isadora e Max, entrambi ambienti estremamente ricchi di funzionalità nell'ambito dei media interattivi.
Tali programmi si basano sulla programmazione a nodi o *visual programming* e permettono essenzialmente di gestire, utilizzare e combinare segnali audio e video e segnali provenienti da sensori o dispositivi come Arduino. Per questo progetto la prima scelta è ricaduta su Isadora, principalmente per la stabilità offerta, tuttavia non avendo potuto acquistare una licenza per motivi economici, si è scelto di utilizzare la versione 7 di Max precedentemente acquistata.

Max

Max nasce alla fine degli anni '80 all'interno del prestigioso IRCAM, prevalentemente per l'utilizzo in ambito musicale, sebbene negli anni è stato potenziato e viene ormai utilizzato come ambiente grafico interattivo non solo per la musica e l'audio ma anche più in generale per il multimedia. Max è il nome che identifica sia il software che il linguaggio di programmazione utilizzato all'interno del programma.

Si tratta di un linguaggio di programmazione a nodi pressoché interamente grafico e quindi permette un apprendimento più facile rispetto ad ambienti di programmazione standard. Alla base di Max vi sono gli oggetti, ossia quegli elementi dotati ciascuno di una specifica funzione e che di norma accettano degli input e dispongono di uno o più output. All'interno di una *patch* è possibile quindi istanziare diversi oggetti e mediante un'opportuna connessione di questi ultimi è possibile generare una composizione sonora o visiva. Al pari della programmazione orientata agli oggetti, anche in Max gli oggetti presenti in una patch rappresentano istanze di una classe; ad esempio `metro 100`, oggetto che si comporta come un metronomo, generando un `bang` ogni 100 ms, è un'istanza della classe `metro`. Un oggetto è quindi definito da un nome e una serie di argomenti. A seconda della funzionalità dell'oggetto possono essere presenti degli *inlet*, ossia punti di ingresso tramite i quali è possibile ricevere l'output di un altro oggetto, e/o degli *outlet*, ossia punti di uscita che forniscono gli output dell'oggetto. In figura 8.31 sono riportati gli inlet e gli outlet dell'oggetto message.

Il collegamento tra gli oggetti avviene mediante le cosiddette *patchcords*, dei fili virtuali che possono trasportare tre diversi tipi di dati: messaggi, segnali audio e segnali video. Nella figura 8.32 sono riportati alcuni oggetti che usano tali tipologie di dati.

Per quanto riguarda l'elaborazione e la gestione dei segnali video e della grafica 3D, Max mette a disposizione il *pacchetto* Jitter, il quale offre un cospicuo numero di oggetti che permettono la manipolazione delle matrici, strutture di dati che si ritrovano alla base di immagini, video e modelli tridimensionali.

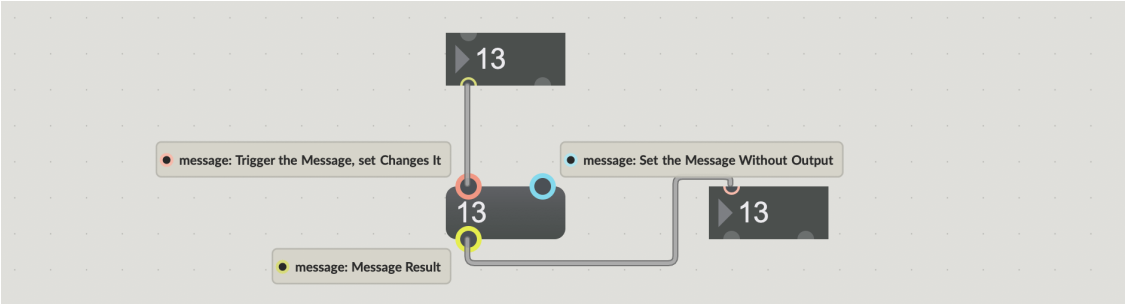


Figura 8.31: L'oggetto `message`, caratterizzato dagli angoli arrotondati, dispone di due inlet e un outlet, con le funzionalità evidenziate in figura.

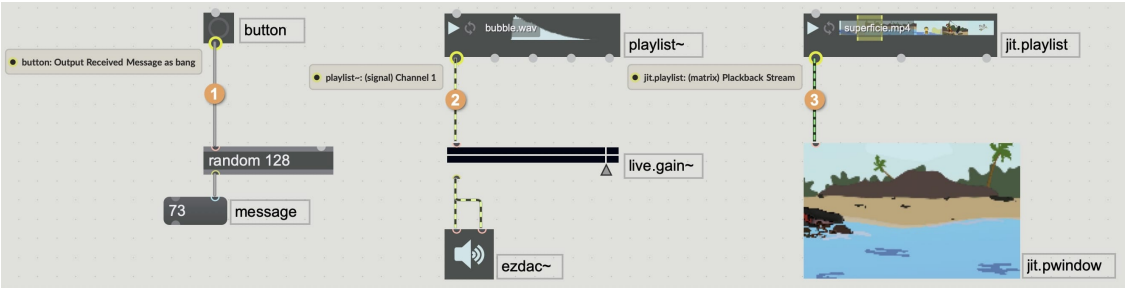


Figura 8.32: Esempio di connessione tra diversi oggetti all'interno di una patch su Max. Per ogni patchcord è evidenziata la tipologia di dati in transito e nello specifico: (1) messaggio (bang), (2) segnale audio e (3) segnale video.

L'implementazione in Max

L'applicativo che gestisce l'interazione dei cubi con il cortometraggio, concretizzato in una patch di Max, deve effettuare le seguenti operazioni fondamentali:

1. In prossimità di uno dei punti di interazione il programma deve inviare il messaggio di avvio del rispettivo gioco previsto, interrompendo la riproduzione del cortometraggio e avviando le indicazioni per lo svolgimento dello stesso.
2. In prossimità della fine del gioco corrente il programma deve ricevere il messaggio che segnala tale evento, quindi riprendere la proiezione del cortometraggio in base all'esito del gioco.

Per quanto riguarda l'implementazione, come riportato nei paragrafi precedenti, il cortometraggio si compone di cinque parti, intervallate da quattro punti di interazione all'interno dei quali viene proposto un gioco. Al lato pratico, ciascuna parte del cortometraggio è stata incorporata in un video, con l'aggiunta finale delle grafiche esplicative che precedono l'inizio del rispettivo gioco sui cubi; tale video verrà in seguito indicato come *blocco*.
Durante la fase di gioco sui cubi, il proiettore riproduce un'animazione in loop,

indicata in seguito come *loopwait*. Alla fine di ciascun gioco, i cubi inviano un messaggio che segnala al computer la fine del gioco e quindi la continuazione del cortometraggio in base all'esito del gioco, che può essere `0` in caso di fallimento e `1` in caso di vittoria. Ad esempio in figura 8.33, rappresentante lo schema a blocchi dell'installazione, il computer riceve il messaggio OSC `/stop 1`, che comporta la riproduzione del blocco 2 nella variante b. Lo schema riportato è puramente esemplificativo, in quanto il primo gioco non ha alcun esito e quindi non comporta una diversa variazione della linea narrativa del cortometraggio. Si hanno quindi tre blocchi relativi alle prime parti del cortometraggio mentre per le due parti, diverse a seconda del risultato del gioco che le precede, sono previsti quattro blocchi. Pertanto la patch implementata in Max deve gestire essenzialmente la *composizione* del cortometraggio sulla base di una serie di messaggi ricevuti dai cubi.

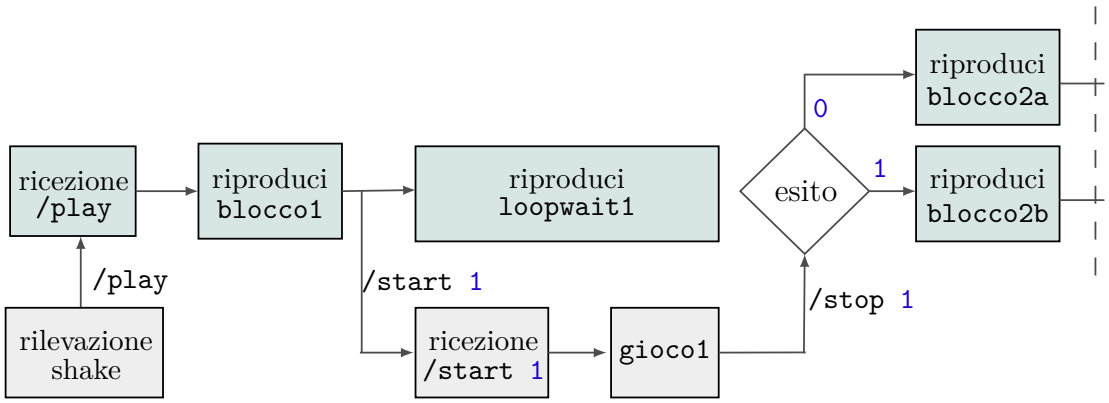


Figura 8.33: Schema esemplificativo alla base dell'installazione.

L'implementazione in Max è riportata in figura 8.34. Come si può notare sono previste due `jit.playlist`, di cui una contiene i blocchi mentre l'altra playlist contiene i loopwait e la grafica per il standby.

L'oggetto `udpreceive 10000` assume la funzionalità di server e quindi rimane in ascolto di eventuali pacchetti UDP sulla porta 10000, i quali vengono passati ai due oggetti `select` sottostanti, che selezionano solo i messaggi con la stringa `/stop` e quelli con la stringa `/play`; l'oggetto `zl mth 1` seleziona invece l'argomento del messaggio OSC e restituisce, solo per i messaggi `/stop`, l'esito del gioco, `0` o `1`. L'oggetto `select /play`, quando trova il messaggio `/play` invia un `bang` dal suo outlet, il quale invia il messaggio `1` alla `jit.playlist` dei blocchi, quindi viene riprodotto il blocco 1.

Alla fine della riproduzione del blocco 1, la `jit.playlist` genera un avviso, `done`, che comporta l'invio dell'identificativo del blocco appena riprodotto; tale messaggio viene quindi indirizzato alla `jit.playlist` dei loopwait, avviando la riproduzione del loopwait 1, e viene infine inviato ai cubi un messaggio `/start 1` tramite l'oggetto

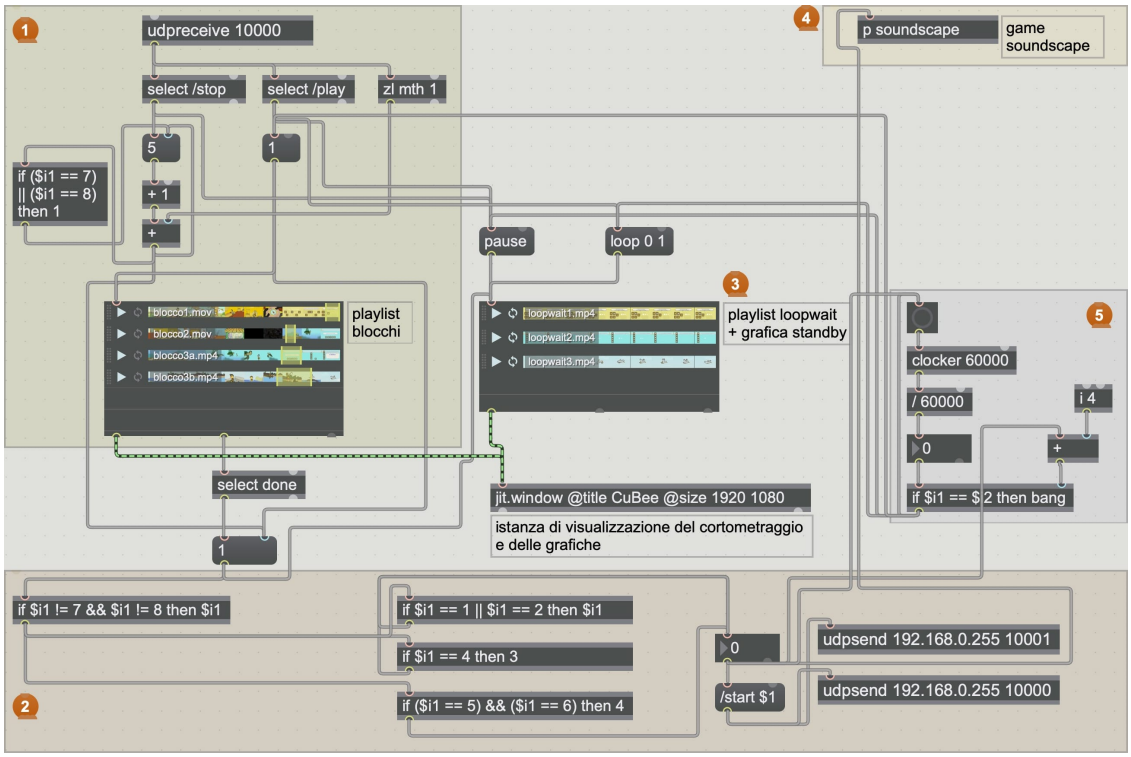


Figura 8.34: Patch Max per l'implementazione della composizione del cortometraggio.

`udpsend` che invia il messaggio al cubo principale, in ascolto sulla porta 10001 e ai cubi secondari sulla porta 10000.
Contemporaneamente alle due operazioni precedenti l'identificativo del gioco in corso viene inviato alla subpatch soundscape, oggetto `p soundscape`, che genera la colonna sonora del gioco, e ad un contatore che in caso di interruzione da parte dell'utente del gioco, inizializza la patch e quindi il sistema si pone in attesa di un nuovo utente.

Al fine di amplificare l'immersione dell'utente durante l'utilizzo dei cubi, si è scelto di sfruttare la dimensione sonora, la quale si articola su due layer: innanzitutto vi è la componente proveniente dai cubi stessi e in secondo luogo vi è la componente proveniente dall'impianto audio dello spazio museale. Si è scelto di utilizzare questa strategia in quanto si è ritenuto importante adottare un diverso approccio a seconda delle dinamiche e della tipologia del suono:

1. *Sound effects*: in determinati contesti risulta più coerente per l'utente identificare la provenienza di questa tipologia di suoni, *event-triggered*, con il punto esatto che ne ha causato la riproduzione, identificabile con un specifico cubo. Alcuni esempi di azioni che causano la riproduzione di un effetto sonoro attraverso lo speaker del cubo sono: lo sblocco del lucchetto in *Unlock the cube*, la

raccolta di una bolla di ossigeno o l'inflizione di un danno in *Ocean Getaway* ecc.

2. *Ambience*: tutti quei suoni che compongono, e caratterizzano, il cosiddetto ambiente del gioco, comprendendo eventualmente la musica, favoriscono maggiormente l'immersione dell'utente nel mondo narrativo del gioco per cui necessitano di una potenza di riproduzione maggiore e di una spazializzazione che non è possibile ottenere mediante gli speaker dei singoli cubi.

L'ambience generato nei punti di interazione, quindi per i diversi giochi, si può definire come *procedural soundscape*[13]: la colonna sonora, evolverà in tempo reale in base ad una serie di regole definite in fase di programmazione, risultando unica per ogni ripetizione dell'esperienza. Al momento l'implementazione in Max dell'algoritmo di generazione della colonna sonora è ancora in corso, tuttavia il principio di funzionamento è il seguente: un background statico viene riprodotto per tutta la durata del gioco, mentre una serie di suoni secondari raggruppati in categorie e opportunamente selezionati, vengono riprodotti diversamente, ossia miscelati in rapporti diversi agendo sull'intensità, riprodotti in ordine casuale e con intervalli di pausa casuali.
L'idea è quella di regolare la colonna sonora mediante una serie di parametri legati all'evoluzione del gioco dell'utente e in base a degli indicatori relativi al clima e all'inquinamento del territorio, come ad esempio la CO2 e il PM10, ottenuti dalle misurazioni effettuate e rese disponibili da Arpa Piemonte. Soprattutto questi ultimi dati possono essere utilizzati applicando un filtraggio sui suoni previsti, quindi a seconda dell'involuzione di un determinato parametro la colonna sonora può diventare più ricca nella parte delle alte frequenze, le quali sono tipicamente percepite come maggiormente sgradevoli.

Capitolo 9

Conclusioni

— Elisa Musolino, Stelian Spulber

La presente tesi è nata con l'obiettivo di inserirsi nella conversazione che si è sviluppata negli ultimi anni intorno agli effetti dei cambiamenti climatici, contribuendo a sensibilizzare più persone possibili, a partire dai più giovani. Tale intento si è concretizzato nella progettazione di un'installazione interattiva da realizzarsi in uno spazio aperto al pubblico. Alla base dell'installazione vi sono due aspetti principali che vengono messi in relazione l'uno con l'altro nell'esperienza proposta: ciò che si è definito come il "virtuale", ossia il mondo narrativo illustrato mediante il cortometraggio di animazione, si lega e interagisce con il "tangibile", rappresentato da una serie di dispositivi che espandono, rendendolo distribuito nello spazio fisico, lo stesso mondo narrativo del cortometraggio.

La prima parte della tesi si è concentrata sugli aspetti coinvolti nella realizzazione di un cortometraggio di animazione, mettendone in evidenza alcune tecniche e strumenti. Dal punto di vista pratico i processi di pre-produzione e produzione del cortometraggio hanno richiesto diverse fasi di lavorazione e tempistiche considerevoli. La definizione della storia è stata ideata in più tempi e la conseguente scrittura della sceneggiatura è stata gradualmente smaltita in quanto ciò che inizialmente si intendeva raccontare risultava avere una durata eccessiva, oltre a richiedere una notevole complessità di lavoro in rapporto al tempo a disposizione.
Nonostante la sceneggiatura sia stata condensata in modo da realizzare e utilizzare soltanto le scene essenziali per trasmettere il messaggio riguardante la tematica definita, non è stato possibile completare l'intero filmato. È stata tuttavia raggiunta una parte sufficiente per sperimentare i primi due punti di interazione, arrivando ad un totale di 6 minuti di animazioni che comprendono sia il cortometraggio sia le grafiche con le istruzioni per l'utilizzo dei cubi. Le fasi iniziali di definizione dei personaggi e degli ambienti, la realizzazione dello storyboard, la scelta dello stile

da adottare e lo studio di un nuovo programma di animazione, Moho 12, nonché il tempo di pratica necessario per la comprensione e l'utilizzo delle funzioni specifiche riguardanti il disegno, la colorazione e soprattutto l'animazione dei personaggi e degli oggetti, sono i principali fattori che hanno influito sull'andamento del lavoro. Inoltre nel corso della produzione del cortometraggio sono state effettuate ulteriori modifiche a seconda dell'andamento delle animazioni e della storia. Complessivamente le animazioni prodotte dovranno essere ulteriormente rifinite, mentre per quanto riguarda la componente sonora del cortometraggio, questa dovrà essere ancora affrontata.

Nella seconda parte del progetto di tesi sono stati affrontati invece gli aspetti riguardanti la progettazione e lo sviluppo di ciò che costituisce la parte "fisica" del sistema CuBee, consistente in cinque dispositivi di gioco caratterizzati dalla forma cubica. La fase di realizzazione di tali dispositivi, detti anche cubi, ha comportato innanzitutto l'approfondimento degli utenti a cui ci si vuole rivolgere e del contesto museale, che costituisce l'ambito di applicazione principale del sistema complessivo. In seguito, sulle tracce definite in fase di ideazione, si è arrivati al primo prototipo funzionante attraverso diversi step.
Il primo passo è stato lo studio della forma e dei dettagli che caratterizzano i cubi, in maniera coerente con gli elementi narrativi e il tema trattato, ma anche la necessità di verificare alcuni programmi di modellazione che rispondessero alla necessità di poter effettuare in modo rapido e snello una serie di operazioni e modifiche sul modello 3D, oltre alla possibilità di collaborazione allo stesso progetto in tempo reale. Le modifiche che hanno riguardato i cubi sono state numerose nel corso del progetto, a causa di cambiamenti in corso riguardanti soprattutto le soluzioni adottate dal punto di vista hardware e quindi relative ai diversi componenti elettronici utilizzati, ma anche modifiche causate da errori di progettazione e relativi alla stampa 3D. In particolare, il processo di stampa dei dispositivi è stato alquanto laborioso e ha richiesto una parte consistente del tempo a disposizione, considerando le esigenze riguardanti non solo la precisione ma anche la qualità visiva del modello stampato. Avendo utilizzato una stampante consumer di fascia bassa sono stati necessari numerosi interventi sia a livello componentistico che a livello di calibrazione. Un ulteriore elemento rilevante è stato il materiale da utilizzare per la stampa dei dispositivi, che ha comportato prove con filamenti in materiali diversi, optando infine per l'utilizzo di un filamento composito in PLA contenente particelle di legno; tale scelta ha richiesto una serie di prove e affinamenti sia del modello del contenitore dei cubi sia delle impostazioni di stampa, causando non poche complicazioni ma portando infine ad un risultato soddisfacente dal punto di vista strutturale ed estetico.
Il secondo passo ha riguardato la scelta dei componenti elettronici da utilizzare all'interno dei cubi, il loro interfacciamento con la scheda Raspberry Pi e infine la programmazione dei dispositivi e dei videogiochi per essi previsti. Sulla base delle

funzionalità definite in fase di ideazione si sono individuate diverse soluzioni e attraverso una serie di verifiche si è infine arrivati ai componenti presenti nel prototipo funzionante corrente. Gli aspetti più rilevanti dal punto di vista della complessità e quindi delle tempistiche necessarie per l'implementazione hanno riguardato indubbiamente la modularità del sistema ma anche la gestione della componente audio dei cubi. Per quanto concerne la modularità, i primi test effettuati hanno confermato la robustezza della soluzione implementata basata su NFC. La componente audio invece, è quella che attualmente necessita di ulteriori approfondimenti in quanto uno dei problemi riscontrati riguarda la presenza di un fastidioso *pop* immediatamente precedente la riproduzione di un file audio, causando l'inutilizzabilità dello speaker; tale problema è stato solo parzialmente risolto attraverso la riproduzione di un rumore costante fuori dalla gamma udibile delle frequenze.
Per quanto riguarda i videogiochi, l'aspetto più delicato e impegnativo è stato quello relativo all'ideazione dei contenuti da sviluppare, in considerazione del pubblico di riferimento, i cosiddetti *nativi digitali*, dell'ambito museale in cui l'esperienza si svolge, e soprattutto in relazione alle tematiche ambientali affrontate. Quest'ultimo aspetto ha infatti richiesto uno studio approfondito per poter essere declinato sotto forma di giochi, nei quali sono stati quindi inseriti elementi educativi ed informativi inerenti a specifici temi ambientali e allo stesso tempo vengono proposti sotto forma di intrattenimento. La nostra scelta è stata quella di progettare una serie di contenuti che fossero coerenti con il mondo narrativo rappresentato nel cortometraggio, in modo tale che l'utente possa vivere l'esperienza *in continuità*. Inoltre tali giochi sono stati studiati in modo da sfruttare tutte le funzionalità dei cubi, come il tocco, lo shake, la componente sonora e soprattutto la modularità; tutto ciò è stato chiaramente inserito in un'ottica che mette al centro l'utente, pertanto si è cercato di tenere in considerazione alcuni aspetti legati alla *Usability* e alla *User Experience*, sebbene alcuni punti siano certamente migliorabili e necessitino anche di una fase di test con gli utenti, prevista come sviluppo futuro.

Infine l'ultima parte della tesi è costituita da ciò che si può raggruppare sotto un'unica definizione, ossia "interazione", intendendo principalmente la partecipazione attiva dell'utente all'interno del sistema CuBee; nello specifico l'interazione con il cortometraggio che è resa possibile mediante l'utilizzo dei cubi in determinati punti della storia, definiti anche "punti d'ingresso".
Dal punto di vista della struttura narrativa dell'esperienza, inizialmente è stato previsto un sistema di interazione basato sull'utilizzo di un'ampia mappa narrativa costituita da molteplici punti di interazione, con una diversa evoluzione della storia per ciascuno di essi. Tale approccio tuttavia avrebbe richiesto un numero piuttosto elevato di contenuti diversi, idealmente pari a $k^n + 1$, con n numero dei punti di interazione e k numero di scelte possibili, risultando adatto per contenuti di breve durata e/o per quei progetti che coinvolgono più persone. Inoltre lo stesso approccio non avrebbe consentito di sfruttare appieno le potenzialità dei cubi realizzati,

comportando un'esperienza poco coinvolgente e ripetitiva per l'utente.
Pertanto si è studiato un modo alternativo di integrare i dispositivi di gioco con il cortometraggio, attraverso la definizione di quattro punti di interazione nei quali i cubi svolgono un ruolo determinante nello sviluppo della narrazione e permettono all'utente di approcciarsi all'esperienza in modalità concettuali diverse: come nel caso di *Unlock the cube*, che dà la possibilità di approfondire una serie di elementi diegetici del cortometraggio; esplorare gli effetti che i cambiamenti climatici possono comportare su determinate specie animali in *Ocean Getaway*; prendere il controllo della situazione e agire, in *Save Puppy*, in cui l'utente può mettere alla prova le proprie abilità schivando i pericoli proposti entro lo scadere del tempo a disposizione; ed infine il caso di *End the Strike*, che assume un forte significato simbolico all'interno dell'esperienza, in quanto rappresenta il più alto grado di interazione dell'utente, il quale viene posto davanti a scelte concrete replicabili nella vita quotidiana che richiedono azioni *collaborative*, esplicitate mediante l'interazione con altri utenti.
Data la complessità e la variabilità dei passaggi e delle funzionalità progettate è stato possibile portare a termine solamente i primi due giochi dei quattro previsti, ma sufficienti a testare alcune funzionalità dei dispositivi di gioco. Pertanto la fase successiva riguarderà lo sviluppo degli ultimi giochi, *Save Puppy* e *End the Strike*, i quali sono stati già progettati e devono essere implementati con Pygame.

Al momento presente ci si è focalizzati sull'ottenimento di un prototipo funzionante con il quale verificare progressivamente una serie di concetti e funzionalità chiave dei dispositivi e del sistema nel suo insieme. Il progetto presenta una notevole complessità e ha posto non poche sfide nella realizzazione, alcune delle quali devono essere ancora affrontate prima dell'organizzazione dell'installazione nello spazio museale. Le difficoltà del percorso affrontato riguardano soprattutto la complessità di sviluppo di un progetto interattivo, combinata con le tempistiche necessarie per la produzione di contenuti originali, la cui realizzazione richiede tempi significativi per ottenere un risultato soddisfacente. Anche in considerazione di quest'ultimo aspetto sono stati progettati soltanto due finali alternativi del cortometraggio, sebbene ad ogni punto di interazione avrebbero dovuto essere previste due diverse evoluzioni del cortometraggio, rendendo sicuramente più interessante l'esperienza ma chiaramente non fattibile con un team di sole due persone, come nel presente caso.
Si è pertanto ritenuto opportuno portare a conclusione il progetto di tesi, proseguendo tuttavia sul percorso che ci porterà a completare i passi necessari per l'organizzazione dell'installazione nel museo. Ciò comporta che l'obiettivo ambizioso, ci sia consentito l'utilizzo di tale aggettivo, che ci siamo prefissati nella fase di definizione dell'idea non è stato realistico e bilanciato con i tempi previsti per una tesi di laurea magistrale, seppur sperimentale.

9.1 Sviluppi futuri

La complessità di un certo numero di aspetti, inizialmente sottovalutati, e gli imprevisti verificatisi hanno comportato non pochi ritardi nell'evoluzione del progetto, pertanto la fase successiva del lavoro si concentrerà sul completamento di quest'ultimo. Nello specifico, per quanto riguarda il cortometraggio, oltre a rifinire alcuni contenuti già prodotti, verranno sviluppate le ultime scene previste dalla sceneggiatura, alle quali seguirà la fase di sound design che ricopre un ruolo chiave. D'altra parte, nel caso dei cubi verranno sviluppati i restanti giochi, *Save Puppy* e *End the Strike*, e sarà ottimizzato il codice sviluppato ad oggi relativo ai primi due giochi. Per quanto riguarda invece la parte di interazione, tra cubi e cortometraggio, verrà completato l'algoritmo alla base della generazione del procedural soundscape riprodotto durante i giochi.
Infine il passo successivo sarà l'organizzazione dell'installazione negli spazi del Museo A come Ambiente. Quest'ultimo aspetto sarà fondamentale per ottenere un riscontro concreto del progetto portato a compimento e valutare quindi l'efficacia raggiunta dal sistema CuBee. Si vuole infatti testare l'attenzione degli utenti partecipanti all'esperienza e la loro disponibilità al confronto e collaborazione con altri utenti per la risoluzione dei problemi e delle sfide poste loro. In particolare si intende valutare se una comunicazione basata sulla gamification e veicolata tramite diversi strumenti interattivi e tangibili, possa facilitare la comprensione di un determinato argomento e incrementare la consapevolezza degli utenti coinvolti nell'esperienza.
L'installazione in uno spazio aperto al pubblico permetterà inoltre di valutare un eventuale proseguimento e adattamento del sistema in contesti diversi da quello considerato. Dal punto di vista dei contenuti realizzati sui dispositivi di gioco, questi sono stati progettati e sviluppati in relazione al cortometraggio, tuttavia il sistema è facilmente adattabile a progetti completamente diversi. D'altra parte, per quanto riguarda il design dei cubi, il guscio di questi può essere facilmente rielaborato, utilizzando forme diverse sulle facce laterali, materiali diversi per la stampa e pertanto è possibile riutilizzare i dispositivi e ciò che costituisce il sistema CuBee.

I dispositivi progettati dispongono di una serie di caratteristiche e funzionalità che ne permettono l'utilizzo anche in ambiti diversi da quelli inerenti alla presente tesi. Essi si prestano infatti all'impiego di un'ampia gamma di contenuti: possono essere utilizzati per la manipolazione di oggetti e la navigazione in ambienti tridimensionali, in ambito sonoro e musicale, sfruttando la loro componibilità e il riconoscimento di una serie di gesture per realizzare diversi soundscape o composizioni musicali ecc.
Nell'ambito dell'apprendimento risulta interessante, ad esempio, l'utilizzo dei cubi come strumento per sviluppare la capacità di individui, bambini e ragazzi,

con difficoltà nell'elaborazione delle informazioni visive e spaziali (abilità visuo-spaziali). Il ragionamento spaziale, associato al feedback visivo e sonoro che si ritrova nell'utilizzo dei cubi può quindi costituire uno strumento di recupero di tali deficit.

In conclusione quindi, le potenzialità dei dispositivi sono notevoli e il nostro obiettivo, completata la prima fase di applicazione del sistema, è quello di approfondirne, qualora possibile, ulteriori usi e applicazioni.

Bibliografia

Capitolo 1

[1] Jacques Aumont e Michel Marie. *Dizionario teorico e critico del cinema*. Lindau, apr. 2007, p. 21.

[2] *Cinema preistorico*. Il Fatto Storico. Giu. 2012. URL: https://ilfattostorico.com/2012/06/16/cinema-preistorico/.

[3] Anna Antonini e Chiara Tognolotti. *Mondi Possibili. Un viaggio nella storia del cinema d'animazione*. Il principe costante, 2008.

[4] Grazia Paganelli. *Reynaud, Émile*. Treccani/ Enciclopedia del Cinema. 2004. URL: http://www.treccani.it/enciclopedia/emile-reynaud_%28Enciclopedia-del-Cinema%29/.

[5] Frank Thomas e Ollie Johnston. *The Illusion of Life: Disney Animation*. Walt Disney Productions, 1981.

[6] Jeffrey A. Okun e Susan Zwerman. *The VES Handbook of Visual Effects: Industry Standard VFX Practices and Procedures*. Elsevier, 2010.

[7] Andy Beane. *3D Animation Essentials*. Wiley, 2012.

[8] NIJI. *WWF Population by Pixel Campaign for WWF Japan*. URL: https://rainbow39.jp/advertising/wwf_pixel/.

[9] Antevenio. *Tipi di contenuti e come vengono processati dal cervello*. Feb. 2019. URL: https://www.antevenio.com/it/blog/tipi-di-contenuti-e-come-vengono-processati-dal-cervello/.

[10] Fabio Marzocca. *Lettura e immagini: le reazioni del cervello. Intervista a Francesco Giordano*. Apr. 2017. URL: https://www.acronico.it/2017/04/19/lettura-e-immagini-le-reazioni-del-cervello-intervista-a-francesco-giordano/.

[11] Teatro Stabile Torino. *Golem*. 2016. URL: https://www.teatrostabiletorino.it/portfolio-items/golem-suzanne-andrage/.

[12] Elissa Blake. *Animation and projections expand theatrical experience.* The Sydney Morning Herald. Mar. 2016. URL: https://www.smh.com.au/entertainment/digital-technologies-of-animation-and-projection-expand-theatrical-experience-20160308-gndxgs.html.

[13] Josef E. Fischer e Kirby I. Bland. *Mastery of Surgery.* Lippincott Williams & Wilkins, 2006, p. 150.

[14] James C. Rosser Jr et al. "The Impact of Video Games on Training Surgeons in the 21st Century". In: *Archives of Surgery* 142.2 (feb. 2007), pp. 181–186.

Capitolo 2

[1] Aditi Bunker et al. "Effects of Air Temperature on Climate-Sensitive Mortality and Morbidity Outcomes in the Elderly; a Systematic Review and Meta-analysis of Epidemiological Evidence". In: *EBioMedicine* VI (2016), p. 258.

[2] Ministero dell'Ambiente e della Tutela del territorio e del mare. *Nota integrativa a legge di bilancio per l'anno 2019 e per il triennio 2019-2021, Tabella N. 9.* 2019.

[3] Commissione Europea. *Le conseguenze dei cambiamenti climatici.* URL: https://ec.europa.eu/clima/change/consequences_it.

[4] WWF. *Il report IPBES prova una perdita di natura senza precedenti.* Mag. 2019. URL: https://www.wwf.it/news/notizie/?47141/Il-report-IPBES-prova-una-perdita-di-natura-ormai-senza-precedenti.

[5] IPBES. *Global Assessment Report on Biodiversity and Ecosystem Services: Summary for Policymakers.* Mag. 2019.

[6] WWF. *Living Planet Report - 2018: Aiming Higher.* 2018.

[7] ISPRA. *Alla scoperta della biodiversità. Strumento didattico del progetto di educazione ambientale "Orientarsi nella biodiversità".* 2010.

[8] S. Shepherd et al. "Extremely Low Frequency Electromagnetic Fields impair the Cognitive and Motor Abilities of Honey Bees". In: *Scientific Reports* 8.1 (2018), pp. 1–2.

[9] M.S. Ahmad e M. Aslam. "Worldwide Importance of Medicinal Plants: Current and Historical Perspectives". In: *Recent Advances in Biology and Medicine* II (2016), pp. 88–93.

[10] Alan Palmer. "Universal minds". In: *Semiotica* 2007 (gen. 2007), pp. 207–209, 213.

[11] James Harris. *The Periodic Table of Storytelling.* 2014. URL: http://jamesharris.design/periodic/.

[12] Tony White. *How to make animated films*. Focal Press, 2009, pp. 235–242, 304–307, 321, 322.
[13] Chris Webster. *Animation. The mechanics of motion*. Focal Press, 2005, pp. 69–71, 109–115, 170, 172, 174.

Capitolo 3

[1] Emmy Rey. *The History of Moho (Anime Studio)*. Smith Micro Software. 2016. URL: https://blog.smithmicro.com/the-history-of-moho-anime-studio.
[2] Smith Micro Software. *Moho. User Manual*. 2016, pp. 3, 4.
[3] Tony White. *How to make animated films*. Focal Press, 2009, pp. 235–242, 304–307, 321, 322.
[4] Chris Webster. *Animation. The mechanics of motion*. Focal Press, 2005, pp. 69–71, 109–115, 170, 172, 174.
[5] Frank Thomas e Ollie Johnston. *The Illusion of Life: Disney Animation*. Walt Disney Productions, 1981.
[6] Richard Williams. *The Animator's Survival Kit*. Faber & Faber, 2009, pp. 102, 103, 107–109.

Capitolo 4

[1] Stefania Serre. *Tassellazioni dello spazio (space tessellation)*. URL: https://areeweb.polito.it/didattica/polymath/htmlS/argoment/ParoleMate/Dic_03/Tassellazione.htm.
[2] Francesco Lamendola. *Il mistero dei poliedri regolari nella teoria platonica del «Timeo»*. Arianna Editrice. Ott. 2011. URL: https://www.ariannaeditrice.it/articolo.php?id_articolo=40565.
[3] *Platone, tutte le opere*. Vol. IV. Newton & Compton, 1997, pp. 591–597.
[4] David Merrill. *Toy tiles that talk to each other*. 2009. URL: https://www.ted.com/talks/david_merrill_toy_tiles_that_talk_to_each_other.
[5] Jeevan Kalanithi e David Merrill. *So long, thank you, we're still here*. Dic. 2014. URL: https://web.archive.org/web/20141229170314/http://farewell.sifteo.com/.
[6] Jeevan Kalanithi e David Merrill. *Sifteo Cubes are now shipping!!!* Set. 2011. URL: https://web.archive.org/web/20120105212240/http://blog.sifteo.com/2011/09/sifteo-cubes-are-now-shipping/.

[7] Jeevan Kalanithi e David Merrill. *Sifteo Cubes "Best in Interface Design"*. Dic. 2011. URL: https://web.archive.org/web/20120308122739/http://blog.sifteo.com/2011/12/fast-company-sifteo-cubes-best-in-interface-design/.

[8] Jeevan Kalanithi e David Merrill. *Vintage Sifteo: A retrospective on our 3-year anniversary*. Feb. 2012. URL: https://web.archive.org/web/20121019131419/http://blog.sifteo.com/2012/02/vintage-sifteo-a-retrospective-on-our-3-year-anniversary/.

[9] Sifteo. *Sifteo - Games*. Set. 2015. URL: https://web.archive.org/web/20150910174132/https://www.sifteo.com/games.

Capitolo 5

[1] Steven D. Eppinger e Karl T. Ulrich. *Product Design and Development*. 5ª ed. McGraw-Hill Irwin, 2012, p. 291.

[2] Andrea Gatto e Luca Iuliano. *Prototipazione rapida: La tecnologia per la competizione globale*. Tecniche Nuove, 1998, pp. 1–3.

[3] J. J. Beaman, John W. Barlow e D. L. Bourell. *Solid Freeform Fabrication: A New Direction in Manufacturing*. Springer, 2014.

[4] John F. Hughes et al. *Computer Graphics. Principles and Practice*. 3ª ed. Addison-Wesley, 2013, p. 341.

[5] Silvia Vizzardelli. *Spazi dell'ascolto e nuova estetica fenomenologica*. Treccani. 2009. URL: http://www.treccani.it/enciclopedia/spazi-dell-ascolto-e-nuova-estetica-fenomenologica_%28XXI-Secolo%29/.

[6] Tatiana Reinhard, Bertier Luyt e Samuel N. Bernier. *Make: Design for 3D Printing*. Make: Community, ott. 2015, pp. 2–8.

[7] Kaziunas Anna France. *Make: 3D Printing: The Essential Guide to 3D Printers*. Maker Media, nov. 2013.

[8] Anmar Frangoul. *Lego to launch sustainable, plant-based plastic pieces*. CNBC. Mar. 2018. URL: https://www.cnbc.com/2018/03/01/lego-to-launch-sustainable-plant-based-plastic-pieces.html.

[9] Eleonora Buiatti. *Forma Mentis. Neuroergonomia sensoriale applicata alla progettazione: Neuroergonomia sensoriale applicata alla progettazione*. FrancoAngeli, 2014, p. 155.

Capitolo 6

[1] Stephen Cass. *Chip Hall of Fame: Intel 4004 Microprocessor*. Lug. 2018. URL: https://spectrum.ieee.org/tech-history/silicon-revolution/chip-hall-of-fame-intel-4004-microprocessor.

[2] Alessio Lana. *Arduino, il chip italiano che ha cambiato l'informatica*. Ott. 2013. URL: https://www.linkiesta.it/it/article/2013/10/06/arduino-il-chip-italiano-che-ha-cambiato-linformatica/16791/.

[3] Rory Cellan-Jones. *The Raspberry Pi computer goes on general sale*. Feb. 2012. URL: https://www.bbc.com/news/technology-17190918.

[4] Rory Cellan-Jones. *A 15 pound computer to inspire young programmers*. Mag. 2011. URL: https://www.bbc.co.uk/blogs/thereporters/rorycellanjones/2011/05/a_15_computer_to_inspire_young.html.

[5] Eben Upton. *Raspberry Pi 3 Model B+ on sale now at $35*. Mar. 2018. URL: https://www.raspberrypi.org/blog/raspberry-pi-3-model-bplus-sale-now-35/.

[6] Rory Cellan-Jones. *Shrinking to Zero: The Raspberry Pi gets smaller*. Nov. 2015. URL: https://www.bbc.com/news/technology-34922561.

[7] BROADCOM. *BCM2835: High Definition 1080p Embedded Multimedia Applications Processor*. Mar. 2015. URL: https://web.archive.org/web/20150316041526/https://www.broadcom.com/products/BCM2835.

[8] Circuit Basics. *Basics of the SPI Communication Protocol*. 2016. URL: http://www.circuitbasics.com/basics-of-the-spi-communication-protocol/.

[9] NXP Semiconductors. *I^2C-bus specification and user manual*. Ver. 6. Apr. 2014. URL: https://www.nxp.com/docs/en/user-guide/UM10204.pdf.

[10] Circuit Basics. *Basics of the I2C Communication Protocol*. 2016. URL: http://www.circuitbasics.com/basics-of-the-i2c-communication-protocol/.

[11] Joy Guo. *Display manufacturers shipped 473 million smartphone displays in Q3 2019*. IHS Markit. Dic. 2019. URL: https://technology.ihs.com/619900/display-dynamics-display-manufacturers-shipped-473-million-smartphone-displays-in-q3-2019.

[12] *Smart watch display shipments total 57 million in Q3, on track to total nearly 200 million by year end*. IHS Markit. Gen. 2020. URL: https://technology.ihs.com/620100/smart-watch-display-

shipments-total-57-million-in-q3-on-track-to-total-nearly-200-million-by-year-end.

[13] Achintya K. Bhowmik. *Interactive Displays*. Wiley, 2015.

[14] Marzio Barbero e Natasha Shpuza.
Che cosa è, come funziona: display a cristalli liquidi (LCD, DSTN, TFT).
CRIT Rai. Ago. 2002.
URL: http://www.crit.rai.it/eletel/2002-2/22-4.pdf.

[15] Guido Masera, Maurizio Martina e M. Nicola.
Slide del corso 'Sistemi elettronici per la produzione e distribuzione'. 2019.

[16] Melissa Hunter.
Understanding LCD Memory and Bus Bandwidth Requirements.
NXP Semiconductors, mar. 2008.
URL: https://www.nxp.com/docs/en/application-note/AN3606.pdf.

[17] Eric Arthur Johnson.
"Touch display–a novel input/output device for computers".
In: *Electronics Letters* I.8 (ott. 1965), pp. 219–220.

[18] Elettronica Open Source. *Gestione di un Touch Screen*. Ago. 2018.
URL: https://it.emcelettronica.com/gestione-di-un-touch-screen.

[19] Texas Instruments. *ADS7846 Touch Screen Controller*. Gen. 2015.
URL: https://www.ti.com/lit/ds/symlink/ads7846.pdf.

[20] Skip Osgood, CK Ong e Rick Downs. *Touch Screen Controller Tips*.
Burr-Brown/Texas Instruments. 2000.
URL: http://www.ti.com/lit/an/sbaa036/sbaa036.pdf.

[21] Wendy Fang e Tony Chang. *Calibration in touch-screen systems*.
Texas Instruments. 2007.
URL: http://www.ti.com/lit/an/slyt277/slyt277.pdf.

[22] Rick Downs. *Using resistive touch screens for human/machine interface*.
Texas Instruments. 2005.
URL: http://www.ti.com/lit/an/slyt209a/slyt209a.pdf.

[23] Rao V. Dukkipati. *Solving Vibration Analysis Problems Using MATLAB*.
New Age International, 2007, p. 42.

[24] TE Connectivity Sensors. *Choosing the right type of accelerometer*.
Lug. 2007. URL: https://www.mouser.com/pdfdocs/choosing-the-right-accelerometer-white-paper.pdf.

[25] Christopher J. Fisher. *Using an Accelerometer for Inclination Sensing*.
Analog Devices. 2010.
URL: https://www.analog.com/media/en/technical-documentation/application-notes/AN-1057.pdf.

[26] Charles Platt. *Make: More Electronics*. Maker Media, 2014.

[27] NXP Semiconductors. *NTAG213/215/216: NFC Forum Type 2 Tag compliant IC with 144/504/888 bytes user memory*. Ver. 3.2. Giu. 2015.
URL: https://www.nxp.com/docs/en/data-sheet/NTAG213_215_216.pdf.

[28] NXP Semiconductors.
PN532/C1: Near Field Communication (NFC) controller. Ver. 3.6.
Nov. 2017.
URL: https://www.nxp.com/docs/en/nxp/data-sheets/PN532_C1.pdf.
[29] NXP Semiconductors. *I^2S-bus specification*. Giu. 1996.
URL: https://web.archive.org/web/20080706121949/http://www.nxp.com/acrobat_download/various/I2SBUS.pdf.
[30] Components101. *Boost Converter: Basics, Working, Design & Operation*.
Gen. 2019. URL: https://components101.com/articles/boost-converter-basics-working-design.
[31] Texas Instruments.
TPS6109x Synchronous Boost Converter With 2-A Switch. Dic. 2014.
URL: http://www.ti.com/lit/ds/symlink/tps61090.pdf.

Capitolo 7

[1] Python Software Foundation. *Cos'è Python*. 2020.
URL: https://www.python.it/about/.
[2] Pierre Carbonnelle. *PYPL PopularitY of Programming Language*. 2020.
URL: http://pypl.github.io/PYPL.html.
[3] Bluetooth SIG. *Bluetooth Core Specification 4.0*. Giu. 2010, pp. 123, 126.
[4] Matt Wright. *The Open Sound Control 1.0 Specification*. 2002.
URL: http://opensoundcontrol.org/spec-1_0.
[5] Tony DiCola. *Adafruit_Python_PN532*. Adafruit. 2015.
URL: https://github.com/adafruit/Adafruit_Python_PN532.
[6] OzzMaker. *BerryIMU*. 2014.
URL: https://github.com/ozzmaker/BerryIMU.
[7] *microbit Device Abstraction Layer*. Lancaster University. 2015.
URL: https://github.com/lancaster-university/microbit-dal.
[8] Jeremy Garff. *rpi_ws281x*. 2014.
URL: https://github.com/jgarff/rpi_ws281x.
[9] Dr0id. *Animation Tutorial*. 2007.
URL: https://dr0id.bitbucket.io/legacy/pygame_tutorial04.html.

Capitolo 8

[1] Maria Grazia Breda. *Formazione e ruolo degli operatori museali e turistici per un piano di rilancio del museo*. Università degli Studi di Catania, 2011, pp. 135, 141–142.

[2] Maria Balshaw et al. *What should our museums look like in 2020?* Mar. 2015.
URL: https://www.theguardian.com/culture-professionals-network/2015/mar/16/museums-in-2020-industry-experts-views.

[3] Claudio Calveri e Pier Luigi Sacco.
Musei & Social Media Sviluppo ed evoluzione dell'interazione utente-museo. DeRev, giu. 2019, pp. 4–5.
URL: https://www.derevworld.com/2019/06/musei-e-social-media-il-report-di-derev-sullo-scenario-italiano/.

[4] Francesco Cervellini e Daniele Rossi. "Comunicare emozionando. L'edutainment per la comunicazione intorno al patrimonio culturale".
In: *DISEGNARECON* (dic. 2011), pp. 48–54.

[5] Francesco Ursini. *La narrazione crossmediale.* Set. 2012.
URL: http://www.treccani.it/magazine/atlante/cultura/La_narrazione_crossmediale.html%5C#.

[6] Henry Jenkins. *Cultura convergente.* 1ª ed. Apogeo, 2006.

[7] *Twine, an open-source tool for telling interactive, nonlinear stories.*
URL: https://twinery.org.

[8] *The Ren'Py Visual Novel Engine.* URL: https://www.renpy.org/why.html.

[9] Mario Ricciardi. *La comunicazione: Maestri e paradigmi.* 8ª ed. Laterza, mag. 2010, p. 83.

[10] Eric Zimmerman. "Manifesto for a Ludic Century". In:
The Gameful World: Approaches, Issues, Applications. MIT Press, 2014, pp. 19–22.

[11] Steffen P. Walz e Sebastian Deterding.
The Gameful World: Approaches, Issues, Applications. MIT Press, 2014, pp. 7, 9.

[12] Lev Manovich. *Il linguaggio dei nuovi media.* Olivares, 2012, pp. 57–59, 62, 63, 79, 128, 130, 260, 264, 265.

[13] Karen Collins. "An Introduction to Procedural Music in Video Games".
In: *Contemporary Music Review* 28 (feb. 2009), pp. 5–15.