

Tesi di Laurea Magistrale

Uno strumento di realtà virtuale immersiva per l'analisi degli incontri e la gestione delle tattiche nel basket

Marco Musto

Relatori

Prof. Fabrizio Lamberti

Ing. Alberto Cannavò



Politecnico di Torino
Ingegneria del Cinema e dei Mezzi di Comunicazione
Sessione di Dicembre 2017

Sommario

La pallacanestro è uno sport in continua evoluzione e nei suoi cento e più anni di storia ¹ si è evoluta velocemente fino a trasformarsi in uno sport caratterizzato da una incredibile componente tattica legata alla elaborazione delle statistiche dei giocatori e al minuzioso studio dei loro movimenti. Quest'ultimo rappresenta ormai una prerogativa indiscussa: qualsiasi squadra che disputi un campionato professionistico ormai prevede uno staff che spesso in numero supera quello dei giocatori in rosa, i cui compiti sono fondamentali al fine di ottenere miglioramenti prestazionali. Figure come il Match Analyst, che si occupa di visionare i video delle partite e selezionare i momenti da rivedere con la squadra o quella dello scout, che crea report statistici a partire dai dati delle partite, sono solo alcuni dei ruoli che al giorno d'oggi sono imprescindibili se si vuole costruire un progetto di squadra funzionale alla crescita e al risultato. La preparazione alla partita è diventata più importante della partita stessa e a fianco di tutta una serie di pratiche volte al miglioramento delle qualità fisiche e atletiche dei giocatori, come le sessioni in palestra e il classico mese di preparazione in ritiro pre-campionato, si ergono allenamenti mirati a infondere ai giocatori i corretti dettami affinché vengano assimilati gli schemi da riprodurre in partita. Questo si traduce in una attenzione focalizzata sempre di più sui dettagli che possono risultare decisivi per costruire il vantaggio da sfruttare durante le fasi di gioco.

L'allenamento in palestra, in cui l'allenatore mostra concretamente le posizioni da occupare in campo, le sessioni video in cui si rivivono gli errori della partita precedente o si studiano i lati deboli e forti del prossimo avversario, gli opuscoli con le statistiche, gli schemi giocati frequentemente dagli avversari o quelli da imparare, i movimenti ripetuti frequentemente dal giocatore chiave avversario e i modi per bloccarlo, la classica lavagnetta cancellabile dell'allenatore, costituiscono le modalità con cui attualmente si prepara una partita durante la settimana che la precede.

L'obiettivo del presente lavoro di tesi è quello di proporre uno strumento per allenatori e giocatori finalizzato alla creazione, gestione e visualizzazione di tattiche di pallacanestro che supera la fruizione della tattica nella classica visualizzazione 2D tipica della lavagnetta o degli schemi su carta e la porta in un contesto di realtà virtuale immersiva, fruibile dal punto di vista del giocatore o di qualsiasi punto scelto nel campo.

Le caratteristiche fondamentali del progetto realizzato sono state implementate grazie al confronto diretto e continuo con Francesco Raho, responsabile del settore giovanile dell'Auxilium CUS Torino, oltre che allenatore nazionale, istruttore nazionale minibasket, formatore per gli allenatori, in più occasioni negli staff delle nazionali giovanili maschili e collaboratore dei centri di alta specializzazione. La necessità primaria - secondo Raho - di un allenatore di dover proporre interventi educativi o correttivi della spazialità tra i giocatori in campo al fine di raggiungere il perfetto equilibrio di rispetto tra spazio d'azione e tempo di gioco, rende la realtà virtuale un campo fertile per proporre uno strumento che offra appunto agli allenatori questa possibilità.

Analizzato il contesto in cui si inserisce questa tesi (Capitolo 1) si è passati ad analizzare lo stato dell'arte (Capitolo 2) degli strumenti tecnologici a supporto

¹La pallacanestro è stata inventata da James Naismith nel 1891 in una scuola di Springfield, Stati Uniti d'America

dell'allenamento nella pallacanestro. Quello che è emerso è che la realtà virtuale ha già fatto il suo ingresso nel mondo dello sport con vari studi che dimostrano l'effettiva efficacia del suo uso al fine di perfezionare il gesto tecnico in vari sport come il rugby, il calcio, il tennis, la danza e anche la pallacanestro. Quello che però si è notato è che a fianco dei software che servono per editare i video, etichettarli e produrre il montaggio da far vedere i giocatori, le applicazioni di creazione tattica, salvataggio e condivisione con i giocatori, rappresentano uno strumento che sempre più allenatori utilizzano nella preparazione delle partite, che però sono limitate alla loro visualizzazione in due dimensioni su schermo nel caso degli schemi e al punto di vista fisso del video, solitamente ripreso dalla tribuna dell'impianto sportivo.

La soluzione proposta dallo strumento realizzato combina le caratteristiche dello schema disegnato sulla classica lavagnetta bidimensionale e quelle della visualizzazione tridimensionale e le estende alla realtà virtuale inserendosi nel solco tracciato dagli studi che confermano che il decision-making in situazioni di gioco frenetiche è allenato in maniera più efficace, rapida e corretta impersonando il proprio giocatore-avatar in uno scenario di allenamento virtuale. Oltre a questo, lo strumento offre la possibilità di elaborare i dati di coordinate spazio-temporali dei giocatori offerte da STATS SportVU ² e di riprodurre in due dimensioni l'azione presa in considerazione. Inoltre, grazie a quanto realizzato in un precedente lavoro di tesi, è possibile elaborare questi dati per riconoscere che tipo di movimento il giocatore sta effettuando e questo permette di riprodurre in realtà virtuale l'azione a partire dalle coordinate spazio-temporali visualizzando correttamente le animazioni base di palleggio, passaggio, blocco e tiro. Inoltre, quanto realizzato in questo lavoro offre la possibilità ai giocatori di essere collegati in tempo reale in una sessione di allenamento in diretta e di visualizzare con un visore di realtà virtuale Cardboard compatibile ³ i movimenti dei giocatori suggeriti dall'allenatore che comanda il tutto dalla sua applicazione lato server.

Per realizzare l'applicazione alla base di questo lavoro di tesi è stato utilizzato il programma di authoring multiplatforma Unity3D ⁴, utilizzando moduli scritti in C# utilizzando Microsoft Visual Studio Code. Le animazioni dei giocatori utilizzate nello scenario virtuale sono state registrate indossando la tuta di mocap Perception Neuron di Noitom ⁵ e utilizzando il programma correlato Axis Neuron, importate prima in Autodesk Maya ⁶ per correggere eventuali imperfezioni durante la registrazione e poi in Unity3D e assegnate ai modelli dei giocatori, anch'essi elaborati con Maya. I requisiti decisi con l'allenatore, l'architettura del programma, la sua implementazione e realizzazione sono discussi nei capitoli centrali di questo testo (Capitolo 3, Capitolo 4, Capitolo 5).

Nei capitoli finali si discute delle implementazioni future che si potrebbero realizzare e dei possibili scenari che potrebbero impiegare la soluzione offerta da questo elaborato di tesi.

²STATS SportVU è il partner ufficiale della NBA per il tracking dei giocatori. Offre i dati di coordinate dei giocatori X,Y, Z in tempo reale delle partite di NBA, il massimo campionato americano di basket.

³<https://vr.google.com/cardboard/>

⁴<https://unity3d.com/>

⁵<https://neuronmocap.com/>

⁶<https://www.autodesk.it/products/maya/overview>

Indice

1	Introduzione	14
1.1	La realtà virtuale	14
1.1.1	Definizione di realtà virtuale	14
1.1.2	Storia della Realtà Virtuale	16
1.2	La pallacanestro	18
1.3	Motivazioni ed obiettivi del lavoro di tesi	19
1.4	Organizzazione del documento	19
2	Stato dell'arte	22
2.1	La tattica nella pallacanestro	22
2.1.1	La preparazione della partita	23
2.1.2	Tecnologie a supporto della preparazione della partita	23
2.2	La realtà virtuale e lo sport	27
2.2.1	La realtà virtuale e la pallacanestro	32
2.2.2	Considerazioni	34
3	Progettazione	35
3.1	Requisiti di progetto	35
4	Tecnologie	37
4.1	Software	37
4.1.1	Unity3D	37
4.1.2	Visual Studio Code	38
4.1.3	AXIS Neuron	39
4.1.4	Autodesk Maya	40
4.2	Hardware	41
4.2.1	Samsung Gear VR	41
4.2.2	Perception Neuron	42
5	Realizzazione	45
5.1	Scelte implementative	45
5.1.1	Architettura del sistema realizzato	46
5.2	Interfaccia per la gestione degli input touch	46
5.2.1	TouchScript per Unity3D	46
5.3	Moduli fondamentali dell'applicazione	48
5.3.1	Gestione giocatori	48
5.3.2	Generazione percorso	49
5.3.3	Generazione contenuti 3D dalle coordinate 2D	51
5.3.4	Gestione del tempo	55

5.3.5	Networking	55
5.4	Utilizzo dell'applicazione	60
5.4.1	Programma per allenatori	62
5.4.2	Programma per i giocatori	67
6	Conclusioni	70
6.1	Considerazioni finali	70
6.2	Possibili implementazioni e lavori futuri	71

Elenco delle figure

2.1	La schermata principale di Dartfish	25
2.2	L'interfaccia principale di FastDraw	26
2.3	La schermata di Basketball Playbook Home	26
2.4	L'ambiente virtuale del campo di rugby usato per il caso di studio [19].	28
2.5	L'ambiente virtuale del campo di pallamano usato per il caso di studio [21].	28
2.6	Un esempio di esercizio rivissuto con STRIVR. Con tecniche di motion tracking sono inserite le istruzioni sullo schema all'interno dell'esperienza immersiva stessa.	30
2.7	Il simulatore immersivo SIDEKIQ nella sua configurazione.	31
2.8	Uno scenario tipico di training di Beyond Sports.	32
2.9	Le statistiche delle percentuali di Ian Mahinmi nel periodo di utilizzo e non utilizzo di STRIVR. "Weekly FT Percentage" rappresenta la percentuale dalla lunetta nella settimana, mentre "Weekly FT Attempts", rappresenta i tentativi effettuati.	33
4.1	Una schermata di Unity.	39
4.2	Esempio di cattura dei movimenti con il software AXIS Neuron e la tuta Perception Neuron	40
4.3	Il layout di default di Autodesk Maya	41
4.4	Principale composizione dei componenti di Samsung Gear VR	42
4.5	Interfaccia del programma AXIS Neuron	43
4.6	Possibili configurazioni della tuta di motion capture Perception Neuron	44
5.1	L'architettura del sistema realizzato.	46
5.2	Schema che raffigura come viene riconosciuta una gesture dalla libreria TouchScript. Fonte: https://github.com/TouchScript/	47
5.3	Esempio di generazione delle frecce in base al trascinamento dei giocatori sul campo	49
5.4	Diagramma del funzionamento dell'inserimento di un keyframe.	50
5.5	Sul campo sono disegnati i path (colore viola chiaro) e i waypoint (colore giallo) creati alla pressione del tasto Play.	51
5.6	Workflow di creazione dei percorsi attraverso i quali si muovono i giocatori	52
5.7	Schema che mostra il funzionamento base del metodo DrillAndPass	53
5.8	Schermata di configurazione della transizione da una animazione all'altra. La barra temporale permette di gestire il passaggio tra le due e <i>Conditions</i> consente di aggiungere i valori booleani	54

5.9	L'Animator assegnato ad ogni giocatore con le animazioni di palleggio, passaggio, tiro, ricezione interconnesse tra di loro dalle condizioni scatenanti.	55
5.10	Questa è la configurazione in cui al Mecanim di Unity si assegnano le ossa del modello creato in Maya.	56
5.11	La struttura di ogni oggetto di tipo Timeline.	57
5.12	Grafica che informa la situazione attuale di rete e permette di iniziare o terminare il collegamento. La dicitura "Controlling Client" significa che si sta utilizzando l'applicazione server, mentre quella client se scritto "Receiving Updates"	59
5.13	Il componente PhotonView con il ViewID scelto e i componenti da tener traccia tra i quali compare il Transform dell'oggetto di cui si vogliono trasmettere i movimenti attraverso la rete	60
5.14	Il componente Photon Animation View e i valori booleani scelti da trasmettere	61
5.15	La barra degli strumenti del programma di creazione di tattiche.	62
5.16	Qui è mostrato un keyframe appena inserito: i giocatori sono mostrati nel loro punto di partenza (mesh trasparente) e arrivo (mesh opaca) con una freccia che descrive il loro movimento	64
5.17	La possibilità di scegliere se il movimento è un blocco o un taglio è permessa da questo pulsante di scelta	64
5.18	Un frame tratto dall'animazione del passaggio.	66
5.19	Un frame tratto dall'animazione di tiro.	66
5.20	Un frame tratto dall'animazione del passaggio.	66
5.21	Visualizzazione dell'esperienza immersiva percepita dal punto di vista del giocatore.	67
5.22	La visualizzazione che viene proposta nel visore di realtà virtuale quando l'allenatore sta modificando le posizioni dei giocatori in campo.	68
5.23	La schermata di inizio dell'esperienza VR dove si può scegliere il giocatore da impersonare nello scenario tridimensionale	68
5.24	Un riassunto di cosa succede contemporaneamente nelle due applicazioni per allenatore e giocatori.	69

Elenco delle tabelle

2.1	Riepilogo dei principali software di creazione tattiche per la pallacanestro in commercio: Creazione tattiche (1), Riproduzione in 2D (2), Riproduzione in 3D (3), Riproduzione in realtà virtuale (4), Sessione multigiocatore live (5),	34
4.1	Principali specifiche tecniche della tuta Perception Neuron di Noitom	44

Capitolo 1

Introduzione

Questo capitolo è dedicato all'introduzione dei due concetti chiave su cui si basa il presente lavoro di tesi, in cui si fondono lo sport della pallacanestro e la tecnologia della realtà virtuale (RV). È fornita inoltre la struttura del documento e gli obiettivi di ricerca nonché le motivazioni che hanno portato a elaborare questo progetto.

1.1 La realtà virtuale

La realtà virtuale è una tecnologia in continua e rapida evoluzione. Descrivere in modo preciso i dispositivi che ne permettono la fruizione, renderebbe obsoleto questo capitolo nel giro di qualche mese.

Per questo motivo si preferisce introdurre prima i concetti fondamentali che sono alla base di questa tecnologia e poi i dispositivi essendo quest'ultimi di gran lunga più soggetti al cambiamento nel tempo.

1.1.1 Definizione di realtà virtuale

La prima cosa che si legge aprendo un dizionario alla voce *virtuale* è: “in contrapposizione a reale”¹. Questo porta alla diretta conseguenza che la definizione “realtà virtuale” è un ossimoro.

Per andare nello specifico, Sherman e Craig nel loro libro *Understanding Virtual Reality* [1] scompongono la definizione e analizzano singolarmente le due parole. Realtà è definita come lo stato o qualità di essere reale. Qualcosa che esiste indipendentemente dalle idee che lo riguardano. Qualcosa che costituisce una cosa vera o reale differentemente da qualcosa che è solo apparente mentre virtuale come essere in essenza o effetto, ma non di fatto.

Questa netta auto-contraddizione però rappresenta il nucleo centrale della definizione di realtà virtuale, che nell'immaginario collettivo è usata per descrivere mondi immaginari ricreati al computer (virtuale) che esistono nella nostra mente (realtà).

Per questo l'espressione composta “realtà virtuale” è stata inserita recentemente nei dizionari e il Merriam-Webster, uno più rinomati d'America, la definisce come “un ambiente artificiale che produce un'esperienza sensoriale attraverso la stimolazione dei cinque sensi causata da un computer e nella quale l'azione dell'utente determina parzialmente quello che succede nell'ambiente”.

¹<http://www.treccani.it/vocabolario/virtuale/>

Secondo LaValle, co-autore di due brevetti Oculus SDK [2] [3] e professore alla University of Illinois, la realtà virtuale è un medium che “induce un organismo, attraverso stimoli sensoriali artificiali, ad attuare un comportamento mirato, mentre l’organismo ha poca o nessuna consapevolezza dell’interferenza”. [4]

Questa definizione porta a dover approfondire quattro concetti chiave.

- Comportamento mirato: questo significa che il fruitore della realtà virtuale vive una esperienza creata dall’autore. Questi comportamenti includono azioni come camminare, volare, guidare un mezzo, relazionarsi con altri utenti ecc...
- Organismo: questa parola non è utilizzata a caso, perché le esperienze di realtà virtuale non si riducono soltanto a persone, ma si estendono a qualsiasi organismo vivente, dal fatto che i ricercatori hanno applicato esperimenti di realtà virtuale anche con esseri animali. [5]
- Stimoli sensoriali artificiali: uno o più sensi dell’organismo fruitore sono ingannati a provare sensazioni dettate dagli input delle stimolazioni artificiali.
- Consapevolezza: durante l’esperienza, l’organismo non si accorge di essere in un mondo virtuale. Questo porta ad accettare come normale il fatto di essere *presenti* in un mondo virtuale.

Cosa intende dunque LaValle, quando parla di *inganno*? Il suo discorso si basa su fondamenta di neurobiologia e più in particolare su studi fondati su come reagiscono alcune parti del cervello durante l’elaborazione di informazioni spaziali. È stato dimostrato infatti che le strutture cellulari che si formano nei processi neuronali - che possono essere descritte in modo simile a un diagramma Cartesiano - scaturiti da esperienze di RV, sono le stesse che si formano durante esperienze reali. [6]

Presenza e immersione

Quanto detto sopra permette di collegarsi a due concetti correlati che definiscono quanto un consumatore di realtà virtuale abbia la sensazione di trovarsi a tutti gli effetti in un mondo reale, in base alle caratteristiche della tecnologia usata e alla capacità di nascondere all’utente la percezione di ciò che è reale e ciò che non è. I concetti di *immersione* e *presenza*.

Immersione

La realtà virtuale, grazie all’evoluzione delle tecnologie che la supportano, riesce ormai a fornire un’esperienza che si basa su elaborazioni grafiche sempre più ad alta risoluzione. Inoltre i tempi di latenza, ossia i tempi che intercorrono tra il gesto compiuto dall’utente nel mondo reale (ad esempio il movimento della testa) e la sua riproduzione nello scenario virtuale, sono arrivati ad essere quasi del tutto minimizzati.

Queste caratteristiche riferite alla tecnologia permettono di definire l’*immersione*, il metro oggettivo con cui si misura la capacità di un sistema di realtà virtuale di riprodurre gli stimoli reali percepiti dai nostri sensi all’interno di un ambiente artificiale in cui si trova l’utente. [7] L’immersione, dunque, dipende in maniera imprescindibile dalle caratteristiche della tecnologia utilizzata, che possono stimolare in maniera più o meno reale i cinque sensi dell’utente.

Presenza

Rispetto all'immersione, che è definita attraverso parametri oggettivi, la presenza è uno stato psicologico ed è dunque chiaramente soggettiva e misura quanto il fruitore dell'esperienza abbia la sensazione "esserci", di trovarsi in un luogo differente, nonostante ci si trovi in un altro. L'ISPR (International Society for Presence Research) ha definito nel 2000 la presenza come uno stato psicologico o una percezione soggettiva in cui, anche se l'esperienza corrente di un individuo viene generata e/o filtrata attraverso la tecnologia creata dall'uomo, parte o tutta la percezione dell'individuo non riconosce accuratamente il ruolo della tecnologia durante l'esperienza. [8]

Molti ricercatori considerano la presenza come una forma di illusione in cui gli stimoli della RV contribuiscono a ingannare la percezione dell'utente che crede di vivere realmente l'esperienza. Hanno, inoltre, distinto quattro forme di illusione che costituiscono la base della sensazione di presenza in una non-esistente realtà riprodotta artificialmente.

- Illusione di essere in uno luogo preciso, che è ottenuta quando gli oggetti e il luogo artificiali sono percepiti come provenienti dal mondo reale. Questo è ottenuto con un buon frame-rate, bassa latenza e ottima calibrazione del device. Un'alta latenza rende l'esperienza non realistica.
- Illusione di impersonificare l'avatar predisposto per l'esperienza, cioè la sensazione dell'utente di avere un corpo nel mondo artificiale. E questo non significa essere rappresentati esattamente dal proprio modello 3D, ma avere un avatar che si muove coerentemente con i propri movimenti. Anzi, aumenta la sensazione di presenza più avere un modello 3D in stile cartoon che si muova seguendo esattamente i movimenti dell'utente, che avere un modello 3D da milioni di poligoni esattamente uguale all'utente che però si muove a scatti.
- Illusione di interazione fisica possibile attraverso la corrispondenza tra la rappresentazione visuale e gli stimoli fisici che ne derivano, come ad esempio un force-feedback generato da un controller in mano all'utente
- Illusione di comunicazione sociale attraverso la comunicazione verbale e non verbale con altri avatar rappresentati da altri utenti collegati o generati dal computer.

1.1.2 Storia della Realtà Virtuale

La realtà virtuale ha subito una evoluzione esponenziale negli ultimi anni, ma le sue origini possono essere tracciate in tempi molto più lontani rispetto ai quelli in cui è quasi del tutto entrata nella quotidianità delle persone.

1800

Lo stereoscopio inventato da Sir Charles Wheatstone nel 1832, che crea l'illusione della tridimensionalità a partire da immagine piatte utilizzando piccole variazioni di angolo, ha preceduto l'invenzione della fotografia. Il dispositivo utilizzava lenti angolate a 45 gradi per riflettere l'immagine nell'occhio. [9].

Circa vent'anni dopo David Brewster, inventa uno stereoscopio di consumo che nel 1856 ha venduto più di mezzo milione di esemplari. La sua forma e il suo design sono molto simili a quelli che molti anni più tardi adotterà il Google Carboard.

1900

Gli anni del ventesimo secolo rappresentano quelli in cui la realtà virtuale ha cominciato a superare il concetto di sola visualizzazione di immagini, concetto già raggiunto nell'800, per arrivare a quello di interazione. Negli anni '60 Morton Heilig brevettò Sensorama, un dispositivo che aveva l'obiettivo di stimolare tutti i sensi, tramite il coinvolgimento non solo di vista e udito, ma anche olfatto e udito. Questo device riproduceva filmati prodotti specificatamente per la fruizione attraverso l'apparecchio che è costituito da una sedia i cui movimenti sono sincronizzati con la riproduzione delle immagini su uno schermo stereoscopico.

L'utente sedeva su una poltrona costruita appositamente con leve e meccanismi di contrappesi per simulare le sensazioni di movimento del corpo e tenendo le mani su un manubrio riceveva feedback tattili. A completare il sistema contribuiva un meccanismo che diffondeva aromi ed odori costituito da un tunnel del vento, utile anche a creare flussi d'aria. [10] Sensorama, tuttavia, non riuscì a trovare fondi di investimento, dunque si fermò alla produzione del suo primo prototipo.

Nel 1965 Ivan Sutherland, considerato uno dei pionieri della computer grafica, introduce i concetti chiave dell'immersione in un mondo simulato dotato di input e output sensoriali. I concetti fondamentali alla base dell'articolo [11] sono:

- il display è un portale che permette all'utente di entrare nel mondo virtuale;
- l'obiettivo è far sì che il mondo virtuale sia il più possibile vicino a fornire esperienze sensoriali percepite come reali attraverso immagini che interagiscono in base alle azioni dell'utente, suoni, odori, gusti e sensazioni tattili reali.

Questi concetti sono alla base di quello che può essere considerato il primo visore di realtà virtuale, inventato da I. Sutherland presso l'Università dello Utah tra il 1965 e il 1968. Il funzionamento del dispositivo si basava su due tubi a raggi catodici, che proiettavano grazie ad elementi ottici le immagini generate dal computer negli occhi dell'utilizzatore. L'obiettivo era permettere la visione di immagini 3D sovrapposte ad oggetti della realtà, con un funzionamento molto simile alle odierne tecnologie di realtà aumentata. Il primo prototipo fu soprannominato "The Sword of Damocles", la spada di Damocle, perché era talmente pesante e grande che per essere usato, i ricercatori hanno dovuto montarlo su di un braccio collegato al soffitto. [12]

Nel 1985, insieme ad altri ricercatori NASA, Scott Fisher sviluppò il primo dispositivo head-tracked HMD (Head Mounted Display) con un campo visivo ampio, chiamato VIVED (Virtual Visual Environment Display). [13] La tecnologia si basava sul design di una maschera da sub con le lenti ottenute da due *Citizen Pocket TV*.

Nel 1985 J. Lanier e T. Zimmerman lasciano Atari e fondando VPL Research, che è la prima azienda a commercializzare prodotti di realtà virtuale, termine coniato per la prima volta nella storia da Lanier, come Data Glove, EyePhone e AudioSphere.

Gli anni '90 sono quelli in cui la realtà virtuale sembra così vicina a dover entrare nella quotidianità delle persone che il magazine Wired nel 1993 predice che nel giro di

5 anni una persona su 10 avrebbe indossato un visore RV durante normali momenti di vita quotidiana. ²

2000

La previsione sopra riportata non si è avverata, e anzi la realtà virtuale non ha saputo a fine anni '90 mantenere le promesse fatte ad inizio decade, tanto che i primi anni 2000 rappresentano un'epoca di crisi, dovuta al fatto che i dispositivi non avevano ancora né una robustezza convincente né una comodità tale da alimentare scetticismo attorno alla realtà virtuale.

Questo velo di scetticismo è diminuito a partire dalla prima decade del XXI secolo, periodo in cui la realtà virtuale è ritornata ad essere circondata dallo stesso entusiasmo degli anni precedenti. Questo è dovuto al progresso tecnologico registrato nel campo dei display, sempre più definiti ed economici e agli studi user-centered svolti a inizio secolo che hanno contribuito a approfondire e migliorare la parte di interazione uomo-macchina e gli studi psicologici che hanno portato a rendere più accurata la riproduzione di contenuti reali in un mondo artificiale.

Esempi di successo in questo campo sono Oculus Rift ³ e HTC Vive ⁴. Il primo è nato come campagna di crowdfunding su Kickstarter di Palmer Luckey nel 2012 fino ad essere acquisito da Facebook nel 2014 e finire sul mercato con una versione consumer nel 2015 dopo due versioni disponibile solo per sviluppatori.

HTC Vive è un headset sviluppato da HTC e Valve Corporation che usa una tecnologia di tracking “room scale” che permette all'utente, previa calibrazione della stanza, di muoversi nello spazio e utilizzare dei controller che sono tracciati per interagire con gli elementi della realtà artificiale. [14]

1.2 La pallacanestro

La pallacanestro o basket (diminutivo di *basketball*) è uno degli sport più spettacolari, dinamici e popolari nel mondo, che richiede un elevato livello atletico e tecnico. L'obiettivo è fare più punti della squadra avversaria facendo canestro, dopo aver catturato, passato, dribblato e tirato la palla. La sua invenzione si deve al Dr. James Naismith che nel 1891 aveva bisogno di una attività da svolgere al chiuso per i suoi studenti dell'istituto Massachusetts Springfield durante l'inverno. Il gioco concepito all'epoca della sua creazione era molto più basilico di quello che si vede oggi – il regolamento originale conteneva solo 7 regole – avendo subito nel tempo differenti modifiche.

L'obiettivo principale della pallacanestro è fare più punti della squadra avversaria facendo canestro, ossia far passare la palla attraverso il ferro. Il campo è un rettangolo di gioco di 15 metri di larghezza e 28 di lunghezza, il pavimento in parquet e le linee che lo delimitano ma non ne fanno parte.

²The Inside Story of Oculus Rift and How Virtual Reality Became Reality: <https://www.wired.com/2014/05/oculus-rift-4/>

³<https://www.oculus.com/>

⁴<https://www.vive.com/us/>

1.3 Motivazioni ed obiettivi del lavoro di tesi

La pallacanestro, a causa della sua evoluzione verso una fase in cui qualsiasi dettaglio della preparazione pre-partita incide in maniera considerevole sulla prestazione finale, ha bisogno sempre più di strumenti all'avanguardia per permettere agli allenatori di inculcare nei giocatori i corretti movimenti durante lo svolgimento di uno schema.

Nel presente lavoro di tesi si vuole studiare come la realtà virtuale possa essere applicata al basket apportando miglioramenti nella fase educativa dei giocatori, come lo si fa in altri sport, come il rugby, il baseball, il calcio, la danza, il tai-chi e il tennis. Questo studio si propone come uno strumento che permetta agli allenatori di educare i propri giocatori a occupare nel campo le posizioni giuste, rispettando le corrette tempistiche di spostamento con o senza palla, con il successo dello schema offensivo o difensivo come obiettivo principale.

L'obiettivo principale è quello di partire dal concetto di schema disegnato su carta e di superarlo insieme a quello di bidimensionalità degli strumenti di coaching già esistenti sul mercato per tablet e di portarlo verso una dimensione non solo tridimensionale, ma anche esperienziale, fornendo all'utente la possibilità di indossare un visore di realtà virtuale per sentirsi all'interno dello schema, viverlo come se fosse all'interno della partita o dell'allenamento in campo. Il presente lavoro vuole porsi come anello di congiunzione tra l'allenatore, lo schema da lui disegnato sulla lavagnetta e la corretta realizzazione in campo. Si ritiene che la possibilità di vivere lo schema tattico direttamente dal proprio punto di vista possa rappresentare un plus che permetta al giocatore di assimilare prima le nozioni proposte dall'allenatore, sia perché l'esperienza diretta può permettere di capire più facilmente i tempi e gli spazi da occupare, ma anche perché questo strumento può rappresentare il miglioramento del concetto per il quale i giocatori devono studiare nei giorni antecedenti alla partita interi quaderni con gli schemi disegnati. Questo potrebbe portare al pericolo di non totale comprensione di ciò che è disegnato o addirittura a non avere il giusto "appeal". "Studiare" in realtà virtuale le tattiche può innescare un meccanismo di *gamification* e rendere maggiormente fruibili i contenuti da assimilare. [15]

1.4 Organizzazione del documento

Questo documento è suddiviso nei seguenti capitoli.

- Capitolo 1: Introduzione

Nell'introduzione è illustrato il contesto su cui si fonda questo lavoro di tesi. Sono presentati i concetti chiave della realtà virtuale e della pallacanestro. Nella parte conclusiva è presentata l'organizzazione del documento insieme ai principali obiettivi alla base di questo lavoro.

- Capitolo 2: Stato dell'arte

In questo capitolo è descritta la situazione che riguarda la tecnologia della realtà virtuale, con un approfondimento agli utilizzi della realtà virtuale e in particolare nello sport.

- Capitolo 3: Tecnologie

Questo capitolo è dedicato alle tecnologie utilizzate nel progetto di tesi, a partire da quelle software per arrivare a quelle hardware.

- Capitolo 4: Progettazione

Il capitolo descrive le fasi di progettazione dell'applicazione realizzata, a partire dal confronto con l'allenatore Francesco Raho, responsabile tecnico del Settore Giovanile della squadra e formatore nazionale per gli allenatori e figura importantissima per la realizzazione di questo prodotto, per arrivare alle scelte delle funzionalità del programma.

- Capitolo 5: Realizzazione

Il capitolo mostra le scelte che sono state intraprese per realizzare il progetto, illustrando in modo specifico come esse abbiano influito per la creazione delle varie parti dell'applicazione.

- Capitolo 6: Conclusioni

Questo capitolo offre uno sguardo d'insieme al lavoro proposto, con alcuni sviluppi futuri che si potrebbero intraprendere per migliorare alcuni dei limiti riscontrati nel capitolo precedente.

Capitolo 2

Stato dell'arte

Per rendere chiaro il contesto in cui si inseriscono i requisiti del progetto di tesi, questo capitolo propone uno sguardo approfondito al concetto di tattica nella pallacanestro e più in particolare agli strumenti didattici utilizzati dagli allenatori. In seguito sono analizzati gli strumenti tecnologici di recente applicazione per la creazione di tattiche destinate all'apprendimento da parte dei giocatori e come la realtà virtuale venga utilizzata nello sport. Ciò che si è notato e che ha portato a iniziare a lavorare su questo progetto è che, sebbene la realtà virtuale sia ultimamente uno strumento sempre più preso in considerazione per l'analisi e la correzione dei movimenti nello sport, la sua applicazione alla tattica nella pallacanestro rappresentava una nicchia in cui inserire i risultati del lavoro.

2.1 La tattica nella pallacanestro

La pallacanestro spiegata agli alunni di una scuola di Springfield dal maestro di educazione fisica James Naismith, il cosiddetto padre della pallacanestro [16], è ben lontana da quella che si può ammirare oggi e nei suoi 126 anni di vita ha subito una infinita serie di evoluzioni che l'hanno vista cambiare: regolamento, ruoli, metodi di allenamento, preparazione di una partita, sono solo una parte dell'universo pallacanestro che nella sua storia è stata sottoposta a una costante evoluzione.

Infatti, la pallacanestro dei tempi contemporanei si discosta ampiamente da quella del suo padre fondatore, soprattutto in quegli ambiti che prima facevano da contorno all'allenamento dei fondamentali tecnici (palleggio, passaggio, tiro, blocco, tagliafuori, ecc.), ma che ora sono rivestono una importanza vitale per il raggiungimento dei successi sportivi. Accanto agli allenamenti focalizzati sul miglioramento delle capacità tecniche dei giocatori, sono diventati imprescindibili tutti quei momenti dedicati allo studio dei movimenti che i giocatori devono effettuare in campo e all'analisi degli schemi delle squadre avversarie.

Andrea Capobianco, allenatore della nazionale femminile e della nazionale U20 maschile, afferma che il suo modello di allenamento integrato, per la formazione di giocatori nei settori giovanili, si basa sull'insegnamento di metodi utili a “risolvere le situazioni di gioco per il bene della squadra con l'ausilio dello strumento tecnico (i fondamentali)”. Questo è dovuto al fatto che ormai la tattica rappresenta un aspetto di cui il basket non si può più privare perché “il livello della scelta dello strumento tecnico adatto per il miglior vantaggio possibile, finalizzato al bene di tutta la squadra. [17] Dunque, la tecnica che senza la tattica non può esistere e

viceversa: due sfere in contrapposizione, ma che senza la loro coesistenza, non porterebbero all'ottenimento del vantaggio, il suo mantenimento e la concretizzazione di esso”.

2.1.1 La preparazione della partita

La parabola temporale della preparazione di una partita inizia molti giorni prima della palla a due ¹ e si esaurisce a soli pochi secondi prima del fischio di inizio con le ultime indicazioni del tecnico e dei suoi assistenti.

La preparazione della partita si articola in più fasi, ciascuna delle quali si differenzia per l'obiettivo prefissato. Nella maggior parte dei casi il primo giorno di allenamento della settimana - solitamente il martedì, dopo un lunedì più dedito al recupero post partita - lo si dedica all'analisi della partita precedente. In questo ambito riveste un ruolo importante la figura del “Basketball Match Analyst”. Il suo compito consiste nell'analizzare i video delle partite della squadra e di selezionare i momenti salienti utili per segnalare all'allenatore i movimenti che hanno riscontrato successo e quelli che invece necessitano di un intervento correttivo. A questo si aggiunge l'analisi dei video della prossima squadra da affrontare, di cui vengono evidenziati i punti forti e quelli deboli, sempre attraverso il lavoro del Match Analyst. Questa analisi viene svolta nei giorni vicini alla partita, mentre durante la settimana l'attenzione è focalizzata più sugli errori individuali della partita precedente, con sessioni video mirate fatte per ogni giocatore che avrà la possibilità di confrontarsi con l'allenatore che gli fornisce un video con tutte le azioni da correggere e quelle che hanno portato un risultato positivo. Nei giorni prossimi alla partita lo staff tecnico fornisce ai giocatori un opuscolo che comprende tutte le informazioni utili per affrontare il match, a partire da un ripasso degli schemi studiati in allenamento nella settimana, per arrivare a quelli degli avversari coadiuvati da statistiche che mettono a fuoco le percentuali realizzative dei giocatori e le loro statistiche migliori (rimbalzi, palle recuperate, zone di tiro ecc.)

2.1.2 Tecnologie a supporto della preparazione della partita

Per preparare le sessioni video citate esistono sul mercato diverse applicazioni che si pongono come obiettivo principale quello di dare la possibilità al Match Analyst di editare porzioni di partita registrare, individuare i vari momenti salienti utilizzando etichette standard che rappresentano l'azione (es. tiro, passaggio, blocco, taglio, fallo, palla persa, palla recuperata, rimbalzo ecc.) ed esportare clip personalizzate in base alle esigenze dell'allenatore, ovvero in base a quali tipi di video si devono mostrare ai giocatori per evidenziare alcune criticità in particolare o movimenti da riprodurre perché hanno generato un vantaggio.

Synergy

Synergy Sports Technology è una compagnia statunitense che crea contenuti video on-demand a supporto dell'universo dell'analytics con l'obiettivo dello scouting nella pallacanestro. La partnership in collaborazione con NBA, WNBA, NCAA Division I

¹La palla a due è un termine tecnico che si riferisce al momento iniziale della partita in cui l'arbitro lancia in aria il pallone al centrocampo e due giocatori delle rispettive squadre contengono il primo possesso della partita

e FIBA permette di avere a disposizione un set incredibilmente vasto di dati riguardo qualsiasi azione di ogni partita di questi campionati. ²

Solitamente i video delle partite elaborate da Synergy sono disponibili dalle 12 alle 24 ore dopo della partita, ma quelle di NBA sono tipicamente disponibili a soli quarantacinque minuti dopo la fine della partita.

Synergy utilizza un sistema di log proprietario che permette di catalogare ogni possesso ³ di qualsiasi partita in una collezione di dati in sintetizzati su diversi tipi di indici, come il tipo di giocata, la situazione di gioco o il diverso comportamento dei giocatori. I fruitori di questo servizio possono accedere a questa mole di dati e creare report su misura in base a campi di ricerca personalizzati in base le esigenze (es. studiare il comportamento di un giocatore/team avversario) e ottenere grafici, tabelle riepilogative e molto altro ancora, il tutto correlato dai collegamenti alle clip video corrispondenti ai dati.

Attraverso lo strumento di editing basato su cloud è possibile creare contenuti condivisibili online che possono essere fruiti tramite l'applicazione per dispositivi mobile.

Dartfish e LongoMatch

Dartfish ⁴ rappresenta uno dei programmi leader nel settore dell'analisi video nello sport fornendo strumenti per pallacanestro, calcio, tennis, football, nuoto, hockey, danza, arti marziali e molti altri ancora. Le sue funzionalità chiave sono le seguenti:

- caricamento e cattura live dei video di più partite;
- possibilità di etichettare frammenti di partita delle azioni chiave per differenziarle in base al loro tipo;
- possibilità di annotare le suddette azioni con un qualsiasi tipo di commento e contenuto grafico, sonoro, testuale;
- generazione di montaggi video e report personalizzati in base all'obiettivo di esso;
- condivisione del montaggio con il proprio gruppo di lavoro.

Secondo Javair Gillett, "Director of Athletic Performance" della franchigia NBA degli Houston Rockets, Dartfish rappresenta il perfetto strumento per creare un ponte tra il dipartimento dedicato al miglioramento delle performance atletiche e quello dello staff che cura lo sviluppo delle tattiche in campo.

In questo campo LongoMatch ⁵ rappresenta un'alternativa a Dartfish e a differenza di quest'ultimo è open-source presentando caratteristiche e un workflow del tutto simili a Dartfish, la cui interfaccia principale è illustrata nella Figura 2.1

²Lo scout è una figura all'interno dello staff dell'allenatore che ha il compito di analizzare i dati delle partite per elaborare informazioni utili per affrontare le partite, migliorare lacune della propria squadra e valutare l'acquisto di altri giocatori

³è chiamato "possesso" un'azione di una partita di basket

⁴<http://www.dartfish.com/basketball>

⁵<http://www.longomatch.com>

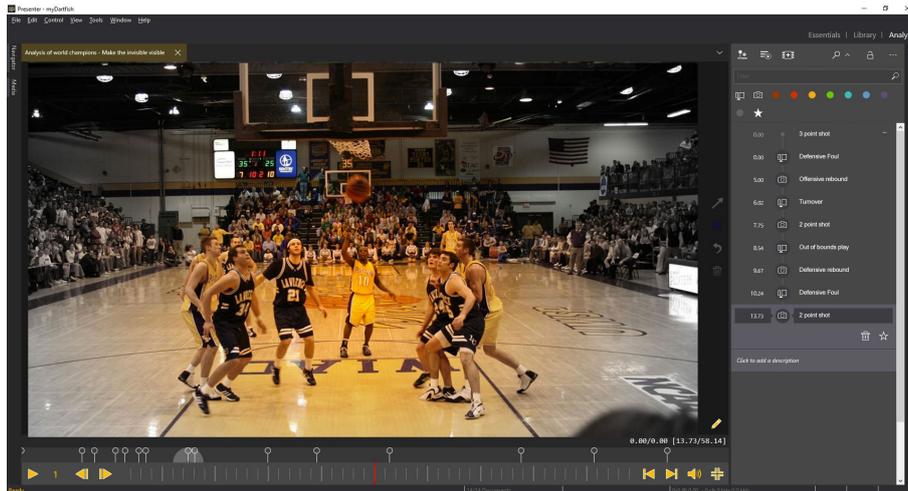


Figura 2.1: La schermata principale di Dartfish

FastDraw

Accanto ai programmi di generazione di report delle partite tramite le annotazioni delle rispettive azioni si affianca tutta una serie di strumenti che permettono agli allenatori di disegnare le proprie tattiche in maniera digitale, superando il tradizionale concetto di schema disegnato su lavagnetta o su carta.

Tra di essi si erge a “standard” sul mercato FastDraw, il cui obiettivo principale è quello di permettere di disegnare, organizzare e condividere con lo staff e i giocatori gli schemi di allenamento o da effettuare in partita.

Le funzionalità principali ⁶ sono descritte di seguito.

- disegno degli schemi utilizzando le annotazioni riconosciute;
- creazione di “playbook” personalizzati;
- accesso a un archivio degli schemi riconosciuti ufficialmente e più giocati;
- condivisione degli schemi con qualsiasi utente FastDraw.

L’allenatore ha a disposizione una interfaccia con in evidenza il campo di gioco in 2D e una barra di strumenti che gli permette di disporre i giocatori in campo e inserire delle frecce per distinguere i vari movimenti chiave come passaggio, movimento con o senza palla, blocco e tiro. Tutto questo viene svolto creando i vari momenti di gioco separati come se fossero fotografie di uno schema che immortalano il movimento chiave. Dunque in output si ottiene un documento che contiene lo schema suddiviso in base ai momenti chiave (es. un passaggio o un taglio) decisi dall’allenatore. Questo può essere inserito in un “playbook”, che consiste in una raccolta di schemi di una squadra o condiviso attraverso il sistema di condivisione tra gli utenti di FastDraw, la cui interfaccia principale è rappresentata nella Figura 2.2.

Basketball Playbook Home

Una valida alternativa a FastDraw è rappresentata da Basketball Playbook Home, prodotto da Jes-Soft ⁷. Questo programma, disponibile per MacOS, Windows, An-

⁶<https://fastmodelsports.com/products/FastDraw-play-diagramming-software>

⁷<http://www.jes-soft.com/playbook/>

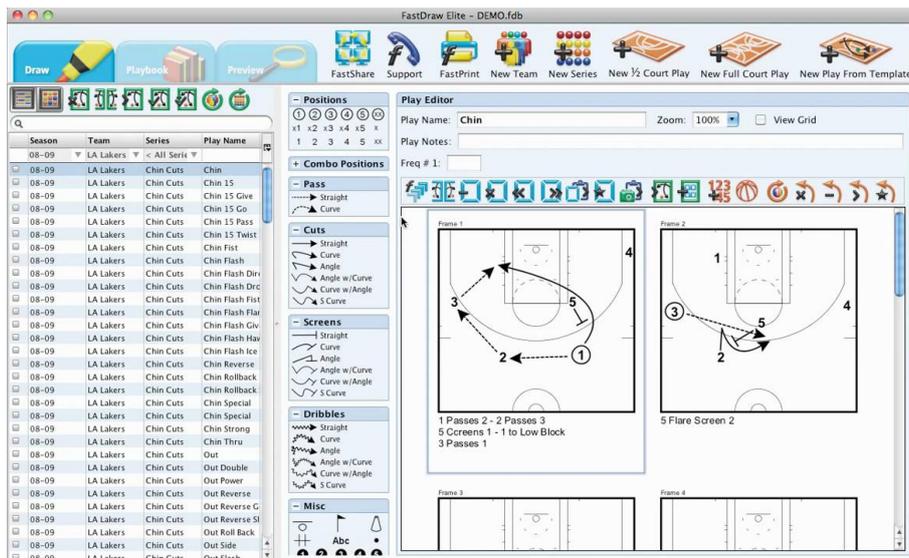


Figura 2.2: L'interfaccia principale di FastDraw

droid e iPhone, si presenta nelle funzionalità simile a FastDraw, però ha il vantaggio di essere gratuito e non disponibile su licenza annuale come invece succede con FastDraw.

Un'altra caratteristica che rende questo prodotto, di cui il layout è mostrato in ?? una eccellente alternativa è la possibilità di fare l'interpolazione in 2D dei movimenti dei giocatori, cosa che non è possibile in FastDraw che ha invece la funzione chiamata "FastBuild" ⁸ che permette di disegnare i frame seguenti a quelli già creati partendo dalle posizioni di arrivo dei giocatori definite dalle frecce inserite per definirne il movimento in una determinata posizione.

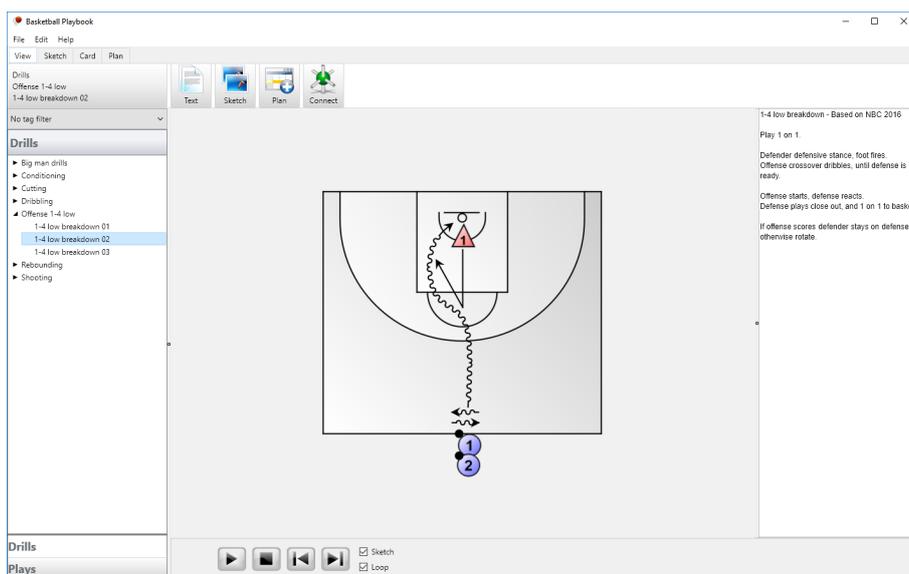


Figura 2.3: La schermata di Basketball Playbook Home

⁸<https://www.youtube.com/watch?v=1n03NeQEMyY>

2.2 La realtà virtuale e lo sport

In questa sezione sono proposti gli esempi più significativi di applicazione della realtà virtuale allo sport, che hanno l'obiettivo di proporsi come strumenti per migliorare le performance nell'ambito dell'educazione di concetti tattici o più semplicemente strumenti per migliorare l'esecuzione cinematica di un gesto fondamentale la cui continua ripetizione porta il giocatore a riprodurlo fedelmente nella maniera corretta. [18]

La realtà virtuale per analizzare le performance sportive

La capacità di discernere le informazioni percettive più rilevanti dal movimento di un avversario è una componente essenziale delle capacità di anticipazione nello sport. Riconoscere in anticipo un movimento di un avversario, permette di generare un vantaggio in una qualsiasi situazione di gioco. Le informazioni necessarie a riconoscere un movimento in anticipo derivano da variabili dinamiche e cinematiche (ad es. la velocità di un giocatore o la sua posizione e orientazione nello spazio) dei movimenti del giocatore avversario. [19]

La visualizzazione di video ha costituito nel tempo un robusto strumento per esplorare le possibili anticipazioni dei comportamenti in una azione. Tuttavia questa maniera di studiare i movimenti dei giocatori ha sempre sofferto del limite di dover sottostare alla posizione della camera, che solitamente è posizionata in un determinato luogo e al massimo cambia la sua angolazione in base a dove si trova in quel momento il fulcro dell'azione.

La realtà virtuale può superare questo limite perché la rappresentazione della posizione dei giocatori risponde a dinamiche generate attraverso simulazioni numeriche che si trasformano in un ambiente immersivo tridimensionale esplorabile in qualsiasi punto spaziale e momento temporale. Questo permette di non dover essere più legato alla posizione fissa della camera ma di potersi muovere in maniera totalmente libera all'interno dell'ambiente sintetico creato su misura. Gli studi di Bideau et al. [19] dimostrano che la realtà virtuale può essere un'ottima via per superare il concetto limite di analisi prestazionale attraverso un video ripreso da un punto fisso. I loro esperimenti si basano su un ambiente virtuale tridimensionale che rappresenta situazioni di gioco del rugby e di pallamano come si vede nella Figura 2.4 e nella Figura 2.5. Nel primo caso di studio viene chiesto ai giocatori di immergersi nello scenario creato ad-hoc - utilizzando un HMD per visualizzare l'esperienza - e di individuare e scegliere utilizzando un game-pad il corretto movimento da effettuare nel momento in cui la riproduzione della parte precedente alla decisione del movimento da intraprendere si blocca e viene posto il quesito.

Nel caso di studio della pallamano il giocatore è immerso in un ambiente CAVE e i suoi movimenti sono catturati utilizzando un sistema di tracking ottico. Davanti a lui compare un attaccante che lancerà una palla nell'ambiente virtuale e questo permette al giocatore di muoversi per cercare di parare il lancio. La combinazione tra i dati di tracking del movimento raccolti e i tempi di reazione permettono di studiare l'efficacia della parata. [20] E più precisamente Bideau e i suoi colleghi hanno dimostrato che un ambiente virtuale immersivo rappresentate un campo di pallamano con un giocatore virtuale che in modo realistico lancia un pallone verso la porta ha portato i portieri a rispondere in maniera simile alle risposte nel mondo reale, precedentemente registrate e catturate con un sistema di tracking ottico. [21]



Figura 2.4: L'ambiente virtuale del campo di rugby usato per il caso di studio [19].

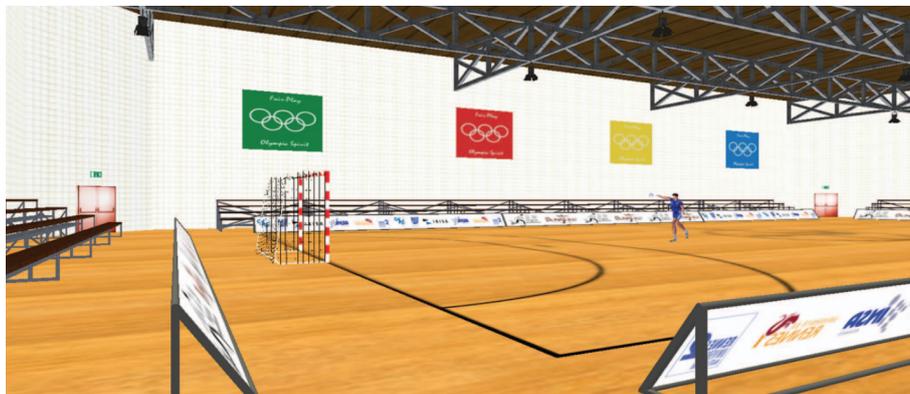


Figura 2.5: L'ambiente virtuale del campo di pallamano usato per il caso di studio [21].

Dunque un ambiente ricostruito al computer permette di controllare quasi in tempo reale fattori che sarebbero impossibili da controllare nel mondo reale, soprattutto perché la realtà virtuale permette ipotetiche infinite ripetizioni del movimento in un ambiente al chiuso, aumentando esponenzialmente i casi di studio applicabili.

La stessa conclusione è stata tratta da una ricerca riguardo l'efficacia della realtà virtuale per studiare e migliorare il comportamento dei portieri nel calcio. Il sistema di Braut et al. [22] è basato su un'esperienza immersiva in cui il portiere si trova realmente in una porta di calcio ma davanti a lui è posto uno schermo retroproiettato 3D in cui è presente l'ambiente virtuale che consiste in una esperienza di un calcio di punizione vissuta dal punto di vista del portiere. L'utilizzatore (il portiere) è tracciato nello spazio attraverso un sistema ottico di cattura dei movimenti e veste degli occhiali 3D per godere delle immagini in stereoscopia sullo schermo e similmente alla ricerca sulla pallamano deve reagire ad uno stimolo che anche in questo caso è rappresentato dalla palla che viene calciata da un giocatore virtuale e che deve essere parata dall'utente. Questa ricerca non solo ha fornito risultati tali per cui è possibile riconoscere la realtà virtuale come un efficace strumento di allenamento, ma ha anche evidenziato come le barriere composte da 5 difensori sono

le più efficaci indipendentemente dalla posizione del calcio di punizione.

Un altro studio che conferma l'efficacia dell'uso della realtà virtuale in prima persona per l'allenamento di situazioni di gioco in cui bisogna prendere una decisione corretta è quello svolto da Petit e Ripoll [23] e poi confermato successivamente da Cardin et al. [24]. Anche in questo caso il giocatore che utilizza questo strumento è sottoposto dover decidere se passare o no la palla durante uno scenario di gioco ricreato in realtà virtuale. In questa ricerca sono stati utilizzati due punti di vista: quello dalla tribuna di uno stadio e quello del giocatore. I risultati hanno confermato che la percentuale maggiore di risposte corrette e effettuate in maniera più rapida è stata effettuata impersonando il giocatore in prima persona e dunque vivendo l'esperienza in prima persona.

A rendere ancora più convincente l'idea che la realtà virtuale applicata all'allenamento di situazioni di gioco in uno sport possa generare risultati più soddisfacenti della classica visualizzazione "piatta" in 2D di una partita da un *fixed point of view* contribuisce il lavoro di Vignais et al. [25] sulle performance dei portieri di pallamano. Questo tipo di ricerca si pone l'obiettivo di definire quale tra le due tecniche di visualizzazione di un'azione possa essere preferita con il fine della migliore analisi visuale dei movimenti. Lo studio si basa su due modalità entrambe le quali necessitano il compimento di un task successivamente ad un input che consiste all'azione che viene bloccata nel momento in cui parte il tiro: il primo consiste nel portiere che deve indicare su uno schermo dove andrà a finire la palla. Nella seconda modalità è il portiere che immerso nell'esperienza virtuale deve muoversi nella corretta direzione successivamente allo stesso tipo di input. Anche in questo caso i risultati tendono verso la realtà virtuale che riscontra prestazioni migliori dei portieri di pallamano professionisti che sono stati studiati che hanno reagito in maniera più corretta, rapida e efficace.

STRIVR

Nata nel 2014 come una start-up di Derek Belch, ex-giocatore e assistente allenatore di football americano all'università di Stanford, STRIVR⁹ si erge ai nostri giorni come azienda leader nel settore della formazione tramite la realtà virtuale proponendo soluzioni di training in realtà virtuale per addestramento in azienda ma soprattutto lo sport. Infatti 17 squadre di NFL, 3 di NBA, 2 di NHL, 1 di MLB - rispettivamente i principali campionati statunitensi di football, pallacanestro, hockey su ghiaccio e baseball - usano i sistemi STRIVR per i loro allenamenti.¹⁰ Ultimamente il sistema è utilizzato anche per allenare gli arbitri, in particolare quelli di NFL.¹¹

L'obiettivo principale dei sistemi di STRIVR è quello di fornire uno strumento che permetta *infinite ripetizioni mentali* degli esercizi nei momenti in cui un atleta è impossibilitato - perché il corpo non può permettere infinite ripetizioni in campo - a provare gli stessi esercizi in campo. I sistemi di training di STRIVR consistono in un ambiente di realtà virtuale immersivo costruito su video a 360 gradi registrati durante reali situazioni di allenamento fruibili attraverso Oculus Rift, HTC Vive e Samsung Gear VR. Questi video mostrano esercizi svolti durante l'allenamento dai

⁹<https://strivr.com/sports/>

¹⁰<https://www.vrfocus.com/2017/04/this-week-in-vr-sport-vr-for-training-in-basketball-and-vr-at-the-gym/>

¹¹<https://www.sporttechie.com/nf-refs-strivr-virtual-reality-training-mental-reps/>

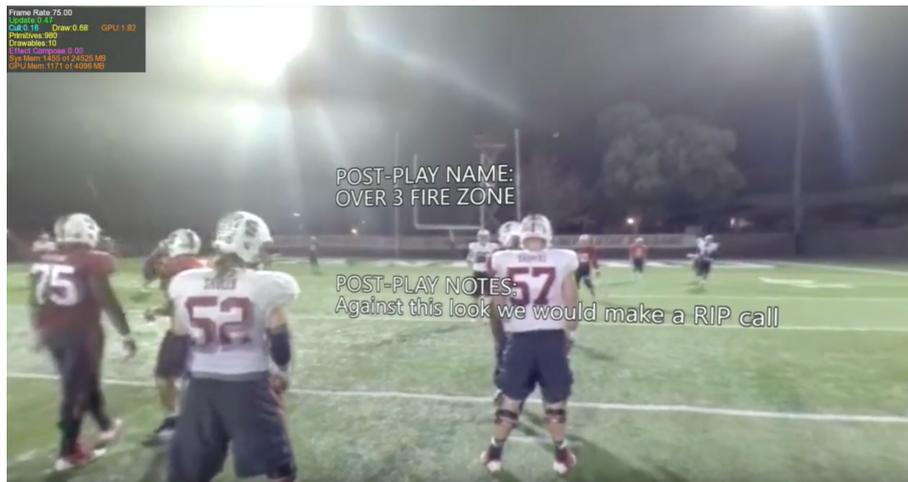


Figura 2.6: Un esempio di esercizio rivissuto con STRIVR. Con tecniche di motion tracking sono inserite le istruzioni sullo schema all'interno dell'esperienza immersiva stessa.

giocatori, registrati dal punto di vista del giocatore, permettendo dunque la possibilità di riprodurre l'allenamento di uno specifico movimento all'infinito (come visibile nella Figura 2.6), aumentando drasticamente l'assimilazione dei concetti chiave. La scelta di utilizzare video reali dal punto di vista del giocatore si basa sul concetto di aumentare in modo esponenziale l'engagement rispetto a una visualizzazione in 2D di una partita di un qualsiasi sport e i risultati portano a una assimilazione dei concetti del circa 37%.

Tuttavia, l'utilizzo di questo sistema negli allenamenti dei Washington Wizards hanno evidenziato i limiti di questo sistema con sport diversi dal football americano. Infatti la moltitudine di posizionamenti statici, le ripetute pause di gioco, l'estrema tendenza a svolgere qualsiasi movimento dettato da uno schema preciso, l'enorme impatto tecnologico che ha portato i quarterback ¹² a poter indossare un microfono per parlare con gli allenatori, rendono i sistemi di STRIVR perfetti per questo sport. E soprattutto idonei alla registrazione dei contenuti durante l'allenamento piazzando specifiche videocamere capaci di registrare video a 360 gradi. I movimenti frenetici dei giocatori e situazioni di gioco di gran lunga più imprevedibili ¹³ che avvengono in sport come la pallacanestro o il calcio rendono più critica la registrazione di questo tipo di contenuti o comunque limitati alla posizione in cui è stato piazzato il particolare rig apposito con le videocamere che producono il video in 3D. ¹⁴ Nonostante questo STRIVR è attiva anche in questi campi e sta portando avanti progetti anche in campo calcistico come dimostrano gli esperimenti fatti presso il STRIVR Lab in collaborazione con la DFB, la federazione calcistica della nazionale tedesca, su situazioni di palla da fermo come punizioni, calci d'angolo e rigori. ¹⁵

¹²uno dei ruoli svolti in una formazione tipo di football americano

¹³<https://washingtonwizardsblog.monumentalsportsnetwork.com/2015/09/30/virtual-reality-hits-the-nba>

¹⁴<https://www.youtube.com/watch?v=ca2S9NCPZEM>

¹⁵<https://www.sporttechie.com/strivr-and-german-football-association-dfb-form-vr-partnership/>



Figura 2.7: Il simulatore immersivo SIDEKIQ nella sua configurazione.

EON Sports

EON Sports, nata nel 2013, è una suddivisione di EON Reality, multinazionale che si occupa di realtà virtuale e offre applicazioni per il training in ambito sportivo oltre a servizi di intrattenimento sempre di carattere sportivo, come ad esempio la trasmissione in diretta della March Madness di NCAA, i playoff del campionato di pallacanestro universitario in USA.¹⁶

Tra i lavori di EON Sports merita citazione SIDEKIQ [26], in Figura 2.7, un simulatore in realtà virtuale di situazioni di gioco di football americano, che a differenza di STRIVR sono realizzate interamente in grafica computerizzata. Il sistema mette a disposizione la possibilità di allenare situazioni di gioco del football con un alto grado di fedeltà del movimento, con un totale di 50 animazioni differenti che possono svolgere i giocatori. Il rendering dell'esperienza è fatta attraverso un 4-wall CAVE per permettere sessioni immersive di allenamento, ma è disponibile anche per visori di realtà virtuale, essendo già stato testato con Oculus Rift. L'esperienza si compone di situazioni di gioco create appositamente da un allenatore professionista: il giocatore è sottoposto a schemi che in un determinato si fermano e richiedono il suo intervento per decidere quale strategia seguire e in base alle risposte giuste o sbagliate, viene elaborato un report del risultato.

Un altro prodotto creato da EON Sports è il "W.I.N. Series" che si propone di fornire uno strumento per l'allenamento professionale di atleti di baseball sfruttando la tecnologia della realtà virtuale. La visualizzazione dell'esperienza avviene attraverso un HMD o un CAVE e i movimenti del giocatore sono tracciati attraverso un sistema di cattura ottica. Questo permette di far provare al giocatore un realistico lancio da parte di un avversario (posizionato di fronte nell'ambiente virtuale) e di rispondere coerentemente tenendo in mano una mazza da baseball, anch'essa tracciata. Questa simulazione permette di, attraverso i dati di velocità, angolazione e precisione dei movimenti del giocatore, fornire un immediato feedback di come la palla è stata colpita e dunque dare al giocatore uno strumento con cui riproporre i movimenti all'infinito al fine del miglioramento del gesto.¹⁷

¹⁶<https://techcrunch.com/2017/03/21/heres-how-to-watch-ncaa-march-madness-in-vr/>

¹⁷<https://www.eonreality.com/press-releases/eon-sports-vr-provides-state-art-baseball-training->

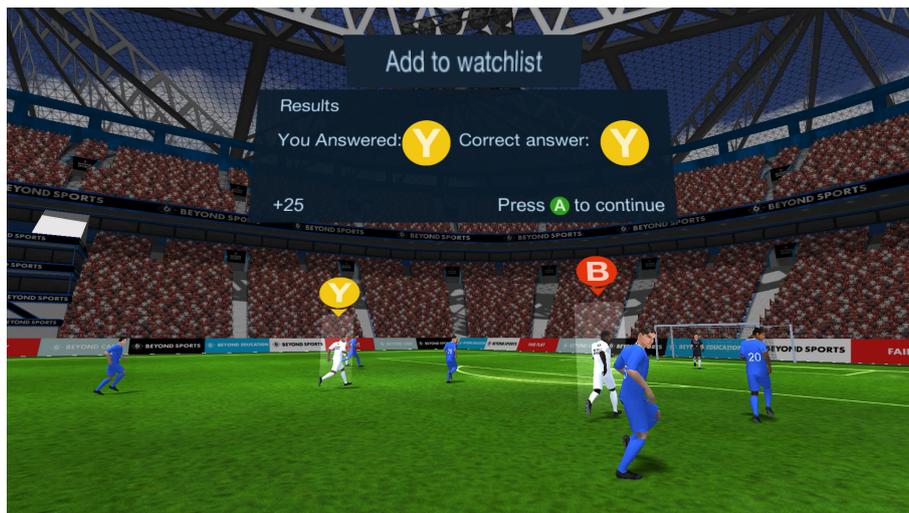


Figura 2.8: Uno scenario tipico di training di Beyond Sports.

Beyond Sports

Un'altra realtà di successo nel campo della realtà virtuale applicata allo sport è Beyond Sports, azienda olandese, che fornisce strumenti per il training sportivo nel calcio con un occhio particolare alla riproduzione in realtà virtuale di intere partite, come dimostra l'accordo con FOX Sports Eredivisie, la tv ufficiale del campionato di calcio olandese, per dare la possibilità ai fan di rivedere sull'applicazione ufficiale tutti i gol del campionato da qualsiasi punto di vista, sfruttando l'elaborazione dei dati di tracking dei giocatori che sono tramutati in grafica tridimensionale. Questa stessa tecnologia è utilizzata alla base degli strumenti di Match Analysis e VR Training, fiori all'occhiello della compagnia olandese che ormai da qualche anno a questa parte rinnova accordi di collaborazione con società di calcio blasonate come Arsenal, Stoke City del campionato inglese e AZ Alkmaar e PSV del campionato olandese. La figura di seguito mostra un esempio di scenario di training in cui l'utente deve scegliere durante l'esperienza di gioco la giusta azione da compiere.

2.2.1 La realtà virtuale e la pallacanestro

Seppur l'intervento della realtà virtuale nella pallacanestro sia ancora un territorio non ancora esplorato totalmente, ci sono degli esempi che meritano di essere presi in considerazione come punto di partenza di questo lavoro di tesi. Ad esempio il prodotto di STRIVR, già citato, che dà la possibilità di immergersi in un campo di basket per rivedere i filmati tridimensionali registrati con camere ad-hoc, è un ottimo strumento per fornire ai giocatori i riferimenti dei corretti movimenti da riprodurre o di quelli sbagliati da evitare. Il suo limite è quello che purtroppo non riesce a superare il concetto di staticità del rig della camera tridimensionale e dunque il movimento del giocatore in parti del campo non coperte dalle riprese non possono essere successivamente esplorate tramite il visore di realtà virtuale.

Un suo utilizzo concreto e che ha portato risultati riscontrati in modo netto è il frutto del lavoro fatto nella stagione 2016-2017 dallo staff tecnico degli Washington Wizards per incrementare le percentuali di tiro libero di Ian Mahinmi. [27] STRIVR

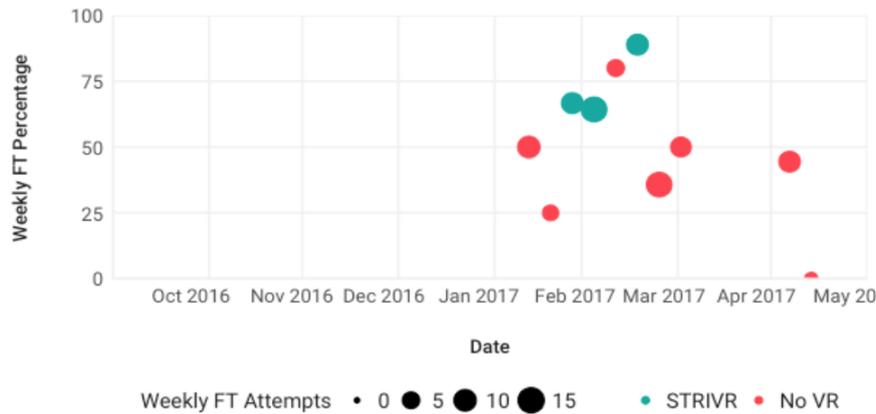


Figura 2.9: Le statistiche delle percentuali di Ian Mahinmi nel periodo di utilizzo e non utilizzo di STRIVR. “Weekly FT Percentage” rappresenta la percentuale dalla lunetta nella settimana, mentre “Weekly FT Attempts”, rappresenta i tentativi effettuati.

ha sviluppato uno strumento per fornire a Mahinmi la visualizzazione del suo movimento quando il tiro libero da lui tirato aveva esito positivo. Questo movimento veniva riprodotto attraverso un HMD per immergere nel campo di gioco il giocatore che poteva vedersi nel momento in cui compiva il gesto (corretto) di tiro dalla lunetta¹⁸. La ripetitività di visualizzazione di un movimento all’interno di un ambiente le cui percezioni portano il giocatore a credere di essere nella realtà porta a immagazzinare le informazioni motorie del gesto da riprodurre in modo corretto. Questo è sottolineato dall’incremento netto delle prestazioni dalla lunetta di Mahinmi (lo schema in Figura 2.9 riporta le sue statistiche) che nei periodi di allenamento con STRIVR aveva una media del 73.3%, mentre nei momenti della stagione in cui non ha fatto uso dell’applicazione, aveva una media del 41.9%.

Un altro studio, sempre sul miglioramento del fondamentale dei tiri liberi è quello di Covaci et al. [28] che propongono un sistema di allenamento per la pallacanestro in realtà virtuale che sfrutta tecniche di motion capture. Il sistema di realtà virtuale elaborato simula una esperienza di tiro libero e si propone come strumento per l’allenamento senza la necessità dell’ausilio di un insegnante. I risultati degli esperimenti hanno dimostrato che i giocatori, che si sono sottoposti alla prova eseguendo una serie di tiri, hanno ottenuto risultati simili sia sul piano reale che su quello virtuale. Inoltre la maggior parte degli utenti che hanno utilizzato il sistema hanno espresso la loro opinione in maniera positiva, apprezzando il simulatore come uno strumento adatto all’apprendimento della tecnica di tiro e alla stimolazione all’allenamento.

¹⁸La lunetta è la parte del campo di pallacanestro dove il giocatore si posiziona per effettuare i tiri liberi a distanza di 5,8 metri dal canestro

2.2.2 Considerazioni

In Tabella 2.1 sono riepilogati i principali prodotti commerciali che si presentano con l'obiettivo di fornire all'allenatore di pallacanestro uno strumento per la creazione, gestione e visualizzazione delle tattiche da mostrare ai propri giocatori.

Come si può notare dal riepilogo, questo lavoro di tesi si propone di superare il limite della visualizzazione bidimensionale delle tattiche offerta dai programmi già in commercio. Più in particolare, oltre alla classica visualizzazione, si vuole offrire la possibilità ai giocatori di esplorare l'azione dal loro punto di vista e di vivere lo schema come se fossero in quel momento in campo. Questo si basa sugli studi citati che affermano che fare esperienza di uno schema tattico in uno scenario virtuale dal proprio punto di vista migliora le capacità di decision-making in situazioni frenetiche, proprio come quelle che si possono vivere durante una partita. In aggiunta, si offre l'opportunità di creare delle vere e proprie sessioni di allenamento in realtà virtuale in diretta, in cui l'allenatore attua i suoi interventi correttivi, che contemporaneamente sono visualizzati dai giocatori attraverso un visore di realtà virtuale.

Tabella 2.1: Riepilogo dei principali software di creazione tattiche per la pallacanestro in commercio: Creazione tattiche (1), Riproduzione in 2D (2), Riproduzione in 3D (3), Riproduzione in realtà virtuale (4), Sessione multigiocatore live (5),

Applicazione	(1)	(2)	(3)	(4)	(5)
Tactics3D	✓	✓	✓	✗	✗
FastDraw	✓	✓	✗	✗	✗
Basketball Playbook Home	✓	✓	✗	✗	✗
LAVORO DI TESI	✓	✓	✓	✓	✓

Capitolo 3

Progettazione

In questo capitolo sono approfonditi gli aspetti legati alla progettazione del sistema di realtà virtuale per l'allenamento delle tattiche.

Nella sezione 4.1 sono elencati i requisiti che sono stati richiesti per la progettazione del software, mentre nelle due sezioni successive 4.2 e 4.3 sono descritti i software e l'hardware utilizzati per la realizzazione dell'applicazione in realtà virtuale.

3.1 Requisiti di progetto

Il progetto descritto in questa tesi è nato in collaborazione con Auxilium CUS Torino Basketball. Le caratteristiche fondamentali del software realizzato sono state implementate grazie al confronto diretto e continuo con Francesco Raho.

L'obiettivo fondamentale consiste nello sviluppo di un sistema di creazione tattiche che unisca alle caratteristiche già presenti nei software comunemente utilizzati, la funzionalità di fruizione della tattica creata non solo nella classica visualizzazione 2D, ma anche nella visualizzazione 3D in realtà virtuale.

Il motivo alla base di questa funzionalità richiesta si fonda sulla possibilità, offerta dalla realtà virtuale, di rendere più efficaci gli interventi dell'allenatore per insegnare o correggere la spazialità tra i giocatori, al fine di raggiungere il perfetto equilibrio di rispetto tra spazio d'azione e tempo di gioco. Questo è un concetto che rientra nei principi fondanti della pallacanestro e secondo Francesco Raho, la visualizzazione bidimensionale sulla classica lavagnetta dei tecnici o sui fascicoli informativi che i giocatori devono studiare nelle sedute tattiche pre-partita non è sufficiente.

Questo è legato a tre motivi principali:

1. l'allenatore può mostrare solo momenti statici che corrispondono a fotografie istantanee della tattica e non ciò che avviene durante il movimento;
2. al massimo può disporre di interpolazione in due dimensioni dei suddetti momenti statici
3. la diretta conseguenza porta ad avere l'assenza della visualizzazione della spazialità da rispettare nell'esecuzione dei movimenti.

Questi tre motivi rappresentano il punto di partenza del software di questo progetto che oltre alla creazione, salvataggio e caricamento di tattiche in 2D, permette

di visualizzare i movimenti interpolati tra una posizione in due istanti di tempo differenti, sia in due dimensioni che in realtà virtuale. Questo tipo di visualizzazione può avvenire in due modalità differenti.

1. La prima prevede che l'allenatore prepari la tattica, la salvi e la condivida con il giocatore che può caricarla nel sistema e fruirlo offline.
2. La seconda modalità è una vera e propria sessione in diretta, in cui ogni giocatore indossa un visore ed è collegato al proprio avatar, in modo tale da vedere in prima persona i propri spostamenti coerentemente con gli spostamenti dei propri compagni e della palla, il tutto sotto la supervisione del tecnico, che tramite l'applicazione sul proprio tablet gestisce i movimenti.

La prima delle due modalità è pensata per offrire al giocatore uno strumento individuale per rivedere gli schemi provati in allenamento e preparati dall'allenatore. La seconda, invece, si basa sulla condivisione nello stesso istante del medesimo contenuto gestito dall'allenatore ed è pensata per quei momenti in cui, durante le sessioni tattiche di preparazione alla partita, ha la necessità di fornire ai propri giocatori le indicazioni sugli schemi da svolgere.

A queste due modalità se ne aggiunge una terza in stretto collegamento con un lavoro di tesi precedentemente realizzato [29], che si basa sulla predizione di passaggi ed eventi di una partita di pallacanestro, a partire dai dati di tracking di coordinate X, Y, Z fornite da STATS SportVU, azienda di punta nel mondo della statistica della NBA. [30]

Questo permette, in linea di principio, di ottenere la riproduzione, prima bidimensionale e poi in realtà virtuale di qualsiasi partita di cui sia disponibile l'elaborazione di questo tipo di dati. Importando un file di testo con le coordinate dei giocatori e gli eventi ricavati, ad esempio, mediante tecniche di machine learning è possibile ricreare la partita in 3d ed eventualmente, cancellare una parte per inserirne una nuova, al fine di creare una tattica correttiva per quel preciso istante di partita.

Questo permette agli allenatori di avere a disposizione uno strumento di tattica che supera il concetto di mera analisi video dei sistemi già presenti in commercio (vedi Dartfish), perché dà la possibilità di generare il contenuto tattico direttamente nel momento in cui, durante una sessione di analisi video coi propri giocatori, è riscontrata una parte di azione che ha bisogno dell'intervento correttivo.

Capitolo 4

Tecnologie

In questo capitolo sono descritte le tecnologie, sia hardware che software, considerate per elaborare il progetto di tesi.

4.1 Software

I software utilizzati per questo progetto sono Unity3D come game-engine per creare l'applicazione scritta interamente in C# con Visual Studio Code, Axis Neuron per registrare, attraverso tecniche di motion capture, le animazioni da inserire in Unity e Autodesk Maya per l'elaborazione di tutti i modelli 3d inseriti nel progetto. Nei seguenti capitoli sono illustrate le funzionalità chiave del progetto.

4.1.1 Unity3D

Unity è un ambiente di sviluppo multiplatforma (utilizzabile sia su Microsoft Windows che su macOS) per la creazione di videogiochi 2D e 3D distribuito da Unity Technologies. Unity è famoso perché permette lo sviluppo di videogiochi su più piattaforme a partire dallo stesso codice scritto. Le piattaforme al momento supportate sono: Android, Android TV, Facebook Gameroom, Fire OS, Gear VR, Google Cardboard, Google Daydream, HTC Vive, iOS, Linux, macOS, Microsoft HoloLens, Nintendo 3DS family, Nintendo Switch, Oculus Rift, PlayStation 4, PlayStation Vita, PlayStation VR, Samsung Smart TV, Tizen, tvOS, WebGL, Wii U, Windows, Windows Phone, Windows Store, e Xbox One.

L'interfaccia grafica è organizzata in modo chiaro e permette all'utente la totale personalizzazione tramite drag and drop dei pannelli. Le sezioni principali sono scritte di seguito.

1. Project: qui è disponibile l'organizzazione in cartelle dei file (nel programma sono denominati *asset*) importati per essere utilizzati nel progetto;
2. Hierarchy: è l'elenco degli asset che sono utilizzati nella Scena corrente;
3. Inspector: permette di visualizzare e modificare gli attributi dell'asset selezionato come posizione, rotazione, ma anche caratteristiche legate al calcolo della fisica real-time, delle collisioni, texture, materiali e proiezione delle ombre.

4. *Scenes*: questo pannello rappresenta la visualizzazione grafica degli oggetti elencati nel pannello *Hierarchy*. L'utente può spostarli, modificare la loro dimensione e rotazione nello spazio 3D. Inoltre può navigare la viewport utilizzato la stessa modalità di interazione comune ai software di modellazione 3D come Maya e Blender. Quando viene caricata una scena, tutti gli oggetti attivi nella *Hierarchy* sono allocati in memoria, dopo che gli oggetti della scena precedente sono distrutti, nonostante sia possibile, attraverso script, mantenere oggetti tra una scena ed un'altra.

Unity si basa sui tre paradigmi fondamentali: `GameObject`, `Component` e `Prefab`. Il primo rappresenta il concetto più importante dell'editor di Unity perché qualsiasi oggetto nella scena è un `GameObject`: dai personaggi, alle luci fino alle camere e gli effetti speciali. Il `GameObject`, però, ha bisogno di proprietà che lo caratterizzino per renderlo un personaggio o una camera o un effetto speciale. Queste proprietà sono conferite dai `components`, la cui combinazione crea un `GameObject` differente. Si può immaginare il `GameObject` come una pentola e i `components` come gli ingredienti alla base della ricetta della scena da creare. [31] Il componente principale di cui non si può mai privare un oggetto è il `Transform` che definisce posizione, rotazione e grandezza secondo il sistema di coordinate XYZ della scena. Oltre al `Transform`, ogni oggetto può contenere componenti built-in di Unity o componenti creati dall'utente tramite script.

Unity supporta nativamente due linguaggi di programmazione:

1. `C#` (C-sharp), un linguaggio di programmazione simile a Java o C++ standard nel settore dei videogames;
2. `UnityScript`, un linguaggio specificamente creato sulla base di JavaScript per essere usato con Unity.

È possibile, inoltre, creare videogiochi utilizzando entrambi i linguaggi di scripting differenti che possono inter-operare tra loro senza nessun problema.

La possibilità di sviluppare la soluzione oggetto del presente lavori di tesi per più dispositivi e il supporto nativo alla realtà virtuale sono i motivi principali che hanno reso necessario ed efficace l'utilizzo di Unity3D. Come linguaggio di programmazione è stato scelto `C#`, perché offre una potenza e versatilità maggiore, una documentazione più accurata e un maggiore numero di utenti che ne fanno uso.

4.1.2 Visual Studio Code

Visual Studio Code è un editor di codice che può essere usato su Windows, macOS e Linux ed è basato sul framework open source Electron. Di default supporta JavaScript, TypeScript e Node.js e consente di aggiungere svariate estensioni per aggiungere il supporto ad altri linguaggi (come C++, C#, Python, PHP, Go) e runtime (come .NET e Unity). Visual Studio Code adotta una classica interfaccia con un *File Explorer* sulla sinistra, dove sono presenti tutti i file che compongono il progetto su cui si sta lavorando e un editor di testo nella parte centrale dove si possono creare e modificare il file del progetto. Tutti gli script creati per Unity sono stati scritti con il linguaggio di programmazione ad oggetti `C#` utilizzando Visual Studio Code come editor.

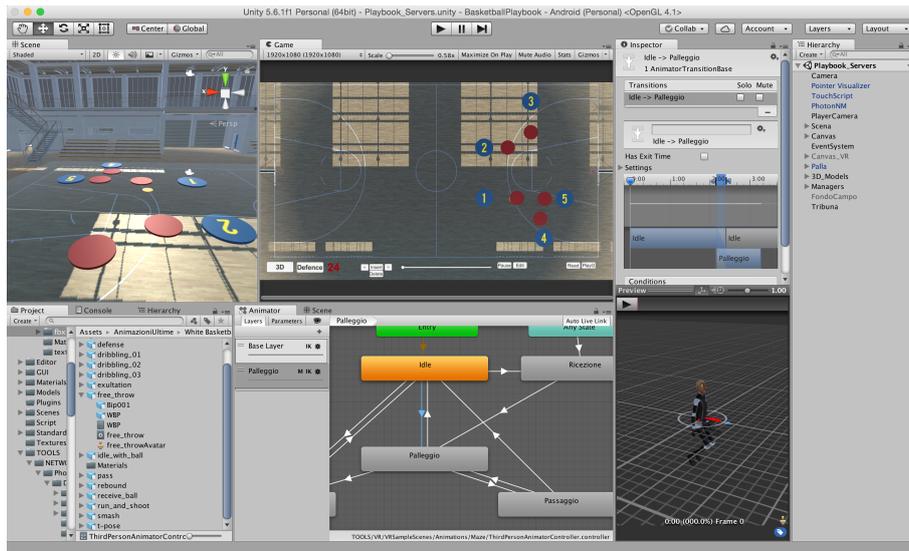


Figura 4.1: Una schermata di Unity.

4.1.3 AXIS Neuron

AXIS Neuron è il software proprietario per la gestione dei dati di motion capture fornito insieme alla tuta Perception Neuron di Noitom. Questa applicazione è lo strumento per la memorizzazione dell'output generato dall'HUB della tuta Perception Neuron, descritta nella sottosezione 4.2.2 e di calibrare i suoi sensori una volta indossata. Una delle funzioni più importanti è quella di trasmettere in tempo reale i dati in .BVH, un formato di condivisione universale dei dati di mo-cap¹ e questo permette l'importazione dei dati in programmi che li elaborano in tempo reale. Allo stesso modo i dati registrati possono essere elaborati nel programma stesso di Noitom e poi esportati in .FBX, formato universale di interscambio di dati 3D², per essere importati nei più comuni programmi di animazione 3D come ad esempio Autodesk Maya.

Le caratteristiche principali di questo software sono:

- gestione semplificata dei sensori;
- sistema basico di calibrazione;
- immunità da interferenze magnetiche;
- personalizzazione della dimensione del corpo dell'avatar 3D;
- layout multi-finestra;
- visualizzazione in 3D in tempo reale dei dati di Motion Capture come si vede nella Figura 4.2;
- possibilità di rielaborare i dati registrati attraverso filtri di smoothing;
- possibilità di modificare i punti di contatto con il suolo;

¹<https://research.cs.wisc.edu/graphics/Courses/cs-838-1999/Jeff/BVH.html>

²<https://www.autodesk.com/products/fbx/overview>

- formato di esportazione .BVH;
- formato di esportazione .FBX;
- fino a 5 mocap collegati contemporaneamente.



Figura 4.2: Esempio di cattura dei movimenti con il software AXIS Neuron e la tuta Perception Neuron

4.1.4 Autodesk Maya

Autodesk Maya, comunemente abbreviato in Maya, è un software di computer grafica che funziona su Windows, macOS e Linux, inizialmente sviluppato da Alias-Wavefront e successivamente da Autodesk. È utilizzato per creare applicazioni 3D interattive come videogiochi, film di animazione, serie TV ed effetti speciali.

Le principali funzionalità sono elencate di seguito.

- Disponibilità di strumenti per modellazione poligonale avanzata, UV mapping, texturing, lighting, rigging e animazione;
- Supporto per uno svariato numero di primitive geometriche come le mesh poligonali, le curve di Bézier, le NURBS e le metaball;
- Possibilità di animare font vettoriali per creare animazioni in motion graphic;
- Simulazione adattiva accurata di fluidi, sistemi particellari, lancio di proiettili e tessuti tramite i vari motori interni integrati;
- Possibilità di gestire l'editing del video in modalità non lineare;

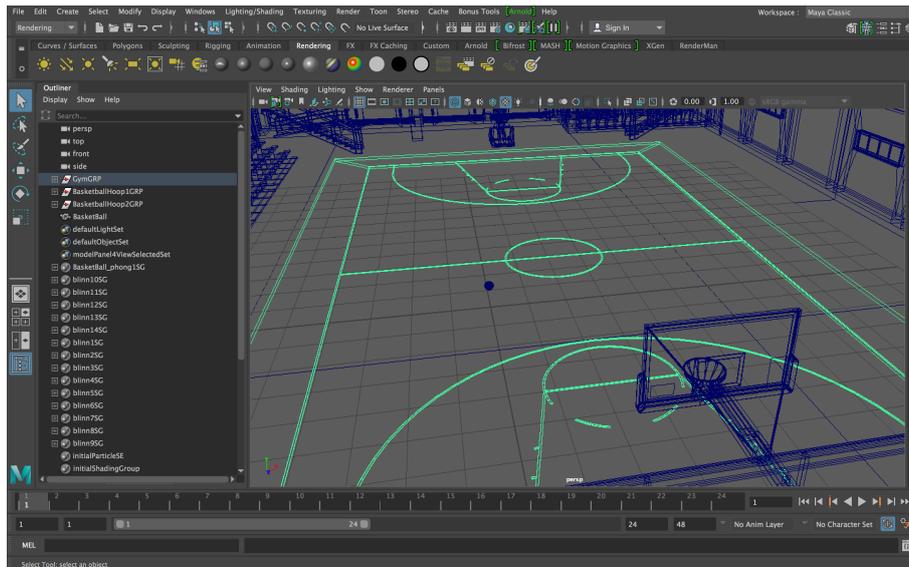


Figura 4.3: Il layout di default di Autodesk Maya

- Possibilità di render con il motore Arnold con altri motori esterni come ad esempio Pixar Renderman;
- Strumenti per creare armature dei personaggi tramite solutori di cinematica inversa o diretta, rigging e skinning e poi animarli grazie al sistema di keyframing su timing integrato;
- Scrivere script programmati nel linguaggio MEL (Maya Embedded Language) per creare moduli aggiuntivi che servono per estendere le funzionalità già presenti nel programma ³.

4.2 Hardware

Qui di seguito sono descritti i principali hardware utilizzati per il lavoro di tesi proposto. Per la produzione dei movimenti dei giocatori riproposti nello scenario di realtà virtuale è stata utilizzata la tuta Perception Neuron di Noitom, una tuta che sfrutta la tecnologia inerziale per catturare i movimenti con un sistema efficace ma al contempo molto più economico dei competitor ottici o magnetici.

L'applicazione realizzata per i coach può essere utilizzata su qualsiasi dispositivo Android, preferibilmente in formato tablet, anche se non si esclude l'utilizzo su computer, perché Unity3D permette di esportare lo stesso progetto su qualsiasi piattaforma supportata senza variare il codice scritto.

Per quanto riguarda l'esperienza in realtà virtuale, essa può essere fruita da qualsiasi dispositivo Cardboard compatibile.

4.2.1 Samsung Gear VR

Samsung Gear VR è un headset per realtà virtuale mobile sviluppato da Samsung Electronics, in collaborazione con Oculus e prodotto da Samsung. Funziona con

³<https://knowledge.autodesk.com/support/maya/learn-explore>

Componenti del dispositivo

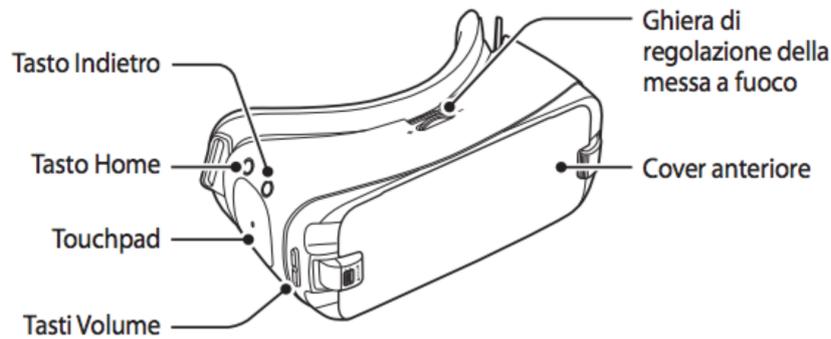


Figura 4.4: Principale composizione dei componenti di Samsung Gear VR

un dispositivo Samsung Galaxy compatibile (come: Galaxy Note 5, Galaxy S6/S6 Edge/S6 Edge+, Galaxy S7/S7 Edge, Galaxy S8/S8+, o Galaxy Note 8). Come può vedersi in figura, il telefono funge da display e processore, mentre il Gear VR funziona come IMU, per il tracking rotazionale. Come è possibile notare in Figura 4.4 l'headset Gear VR include anche un touchpad e un bottone nella parte esterna e un sensore di prossimità per sapere quando il dispositivo è indossato.

4.2.2 Perception Neuron

Perception Neuron è un sistema di cattura del movimento sviluppato nel 2011 dall'azienda cinese Noitom.⁴ È considerato un sistema di Motion Capture adattivo perché permette di tracciare il movimento di tutto il corpo o soltanto di alcune sue parti, secondo le diverse configurazioni offerte. La tecnologia di Perception Neuron è basata su più sensori chiamati Neuron, i quali sono piccoli come una monetina e pesano circa un grammo. Ognuno di essi ospita un IMU (Unità di misurazione inerziale) costituito da un giroscopio a tre assi, un accelerometro a tre assi e un magnetometro a tre assi. Il sistema si basa sull'algoritmo proprietario chiamato "Embedded Data Fusion, Human Body Dynamics and Physical Engine"⁵ che si occupa di ricevere in input i dati dai Neuron, che trasmettono a 60fps o 120fps⁶ e restituire in output un movimento fedele, omogeneo e con la minima latenza. Il sistema, le cui caratteristiche principali sono descritte nella Tabella 4.1, si propone come uno strumento valido per la cattura dei movimenti corporei (per l'uso in videogame, cinema o applicazioni real-time), ma con un costo nettamente inferiore rispetto ai più costosi, e anche più precisi, sistemi di cattura ottica o magnetica.

AXIS Neuron propone una interfaccia grafica di utilizzo molto intuitiva, e la cattura dei dati a partire dai movimenti di chi indossa la tuta avviene in semplici passaggi. Una volta indossata la tuta e collegato l'HUB al computer, la tuta è riconosciuta automaticamente dal programma e calibrati i sensori tramite una procedura guidata, si può iniziare la registrazione cliccando sul pulsante REC. Nella Figura 4.5 si può vedere il layout base del programma che nella parte centrale propone la visualizzazione in tempo reale dell'avatar collegato, nella parte destra è

⁴<https://neuronmocap.com/>

⁵<https://www.noitom.com/index.php/solutions/perception-neuron>

⁶60fps: 19 - 32 Neurons, 120fps: 18 Neurons o meno

presente lo scheletro con i sensori collegati durante l'utilizzo, e nella parte bassa ci sono i pannelli dedicati all'editing dei dati registrati mediante l'utilizzo di filtri di smoothing, di modifica manuale dell'angolo dei giunti dello scheletro e dei punti di contatto dei piedi.

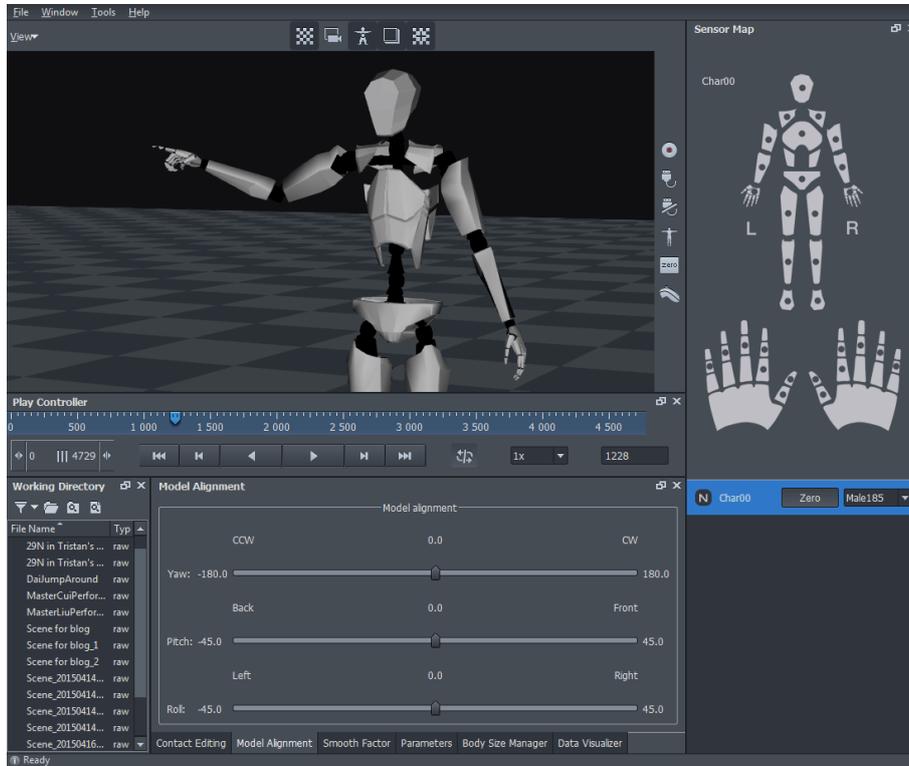


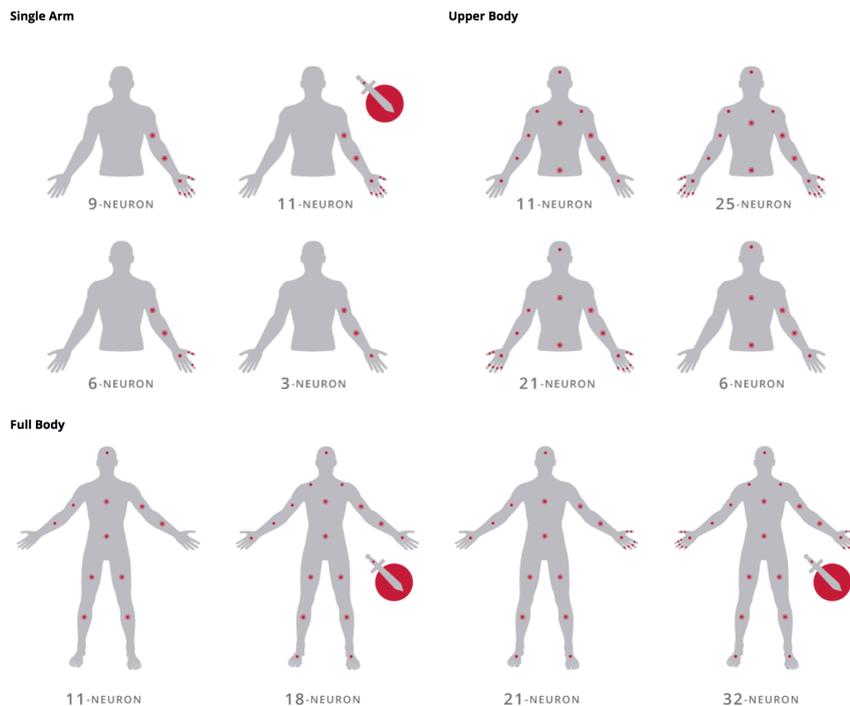
Figura 4.5: Interfaccia del programma AXIS Neuron

Nella Figura 4.6 sono illustrate le diverse configurazioni possibili con la tuta, i quali Neuron (in totale 32) comunicano in tempo reale con il PC via USB o WiFi tramite un Hub collegato alla tuta: si può catturare dalla sola mano, fino ad arrivare ad effettuare il motion capture di tutto il corpo.

Tabella 4.1: Principali specifiche tecniche della tuta Perception Neuron di Noitom

Dimensioni:	12.5mm x 13.1mm x 4.3mm
Intervallo dinamico:	360 deg
Intervallo accelerometro:	circa 16g
Intervallo giroscopio:	circa 2000 dps
Risoluzione:	0.02 deg
Roll:	<1 deg
Pitch:	<1 deg
Yaw angle:	<2 deg
Dimensioni HUB:	59mm x 41mm x 23mm
Max. connected sensors:	32
Output:	USB 2.0, WIFI o Micro-SD Card
Max Output rate:	60 fps con 32 Neurons, 120 fps con 18 Neurons
Energia:	USB

Figura 4.6: Possibili configurazioni della tuta di motion capture Perception Neuron



Capitolo 5

Realizzazione

In questo capitolo sono mostrate le principali scelte implementative che sono state seguite per realizzare il progetto e approfondisce i moduli creati per ottenere le funzionalità richieste. Nella parte finale sono descritte le funzionalità base dell'applicazione realizzata ed uno schema riassuntivo del workflow comune di utilizzo.

5.1 Scelte implementative

Durante la fase di progettazione si era presa in considerazione l'idea di creare un'applicazione che desse la possibilità all'utente di muovere su un dispositivo touch i giocatori e al contempo memorizzare questi movimenti in una struttura temporale per poi vedere il movimento interpolato tra i tempi e gli spazi scelti. Successivamente il programma trasforma questi movimenti bidimensionali in 3D per la fruizione in realtà virtuale. Per fare ciò è stato sviluppato un programma su Unity che ha reso necessario creare degli script ad-hoc per estendere le funzionalità di quelli già presenti di default e soddisfare i requisiti alla base del progetto. È stato scelto un qualsiasi dispositivo tablet come plancia di comando dell'allenatore, dunque il programma è stato esportato per questo tipo di device mentre invece si è deciso di destinare il programma di visualizzazione tattiche dei giocatori a un tipo di device mobile (smartphone) per rendere compatibile la visione attraverso un qualsiasi dispositivo simile al Samsung Gear VR o Cardboard. Quest'ultima scelta è stata effettuata per due ragioni ben precise, discusse di seguito.

1. Samsung Gear VR e Cardboard sono di gran lunga più economici di Oculus Rift e HTC Vive e questo rappresenta un fattore da considerare tenendo conto che una delle funzioni del programma è quella di creare una sessione in diretta di almeno 5 giocatori connessi contemporaneamente. Dare la possibilità di utilizzare il proprio telefono cellulare è stato preso in considerazione come un fattore determinante in favore di Samsung Gear VR invece che dispositivi più costosi.
2. Oltre quanto già detto Samsung Gear VR e Cardboard non hanno l'obbligo di essere connessi con ingombranti cavi e dato che la massima interazione effettuata è legata al movimento della testa, il quale si può rilevare dall'accelerometro e giroscopio del telefono, la scelta è ricaduta su i dispositivi già citati.

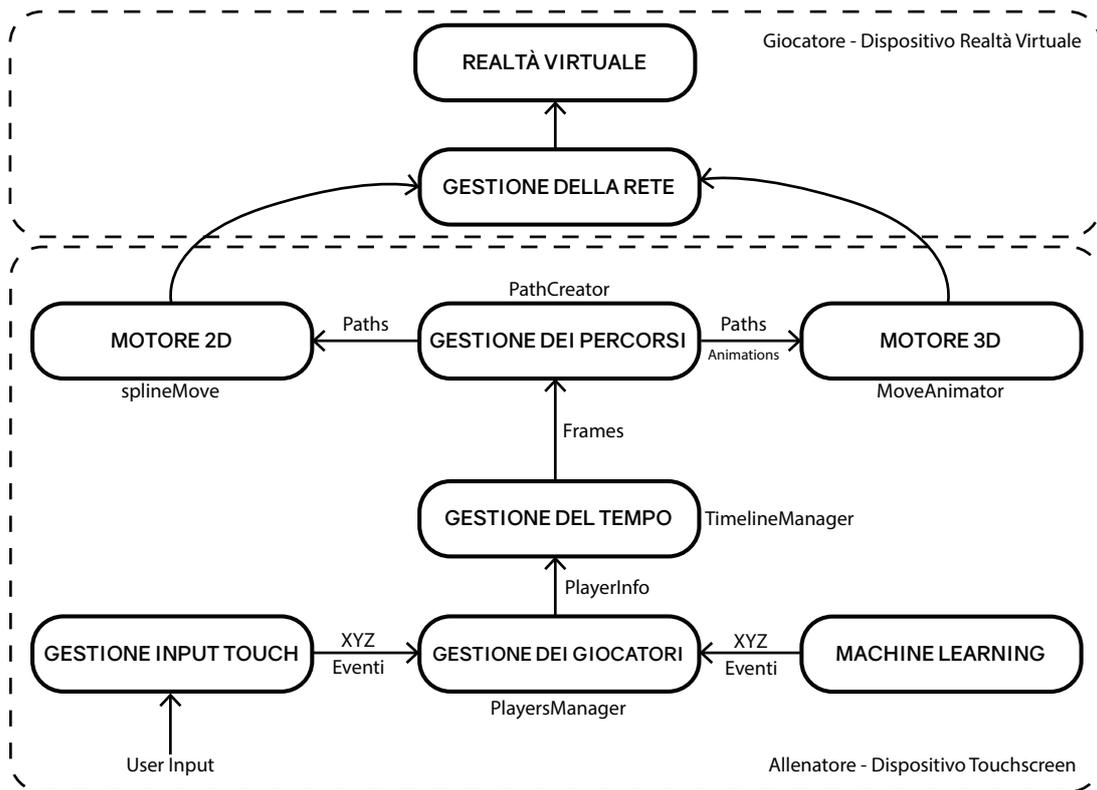


Figura 5.1: L'architettura del sistema realizzato.

5.1.1 Architettura del sistema realizzato

Nella Figura 5.1 uno schema rappresenta l'architettura implementata per l'applicazione realizzata e i vari blocchi fondamentali la cui descrizione segue nella parte centrale di questa sezione.

5.2 Interfaccia per la gestione degli input touch

L'applicazione per gli allenatori fornisce uno strumento con il quale gli allenatori devono creare le loro tattiche muovendo i giocatori sul campo. Per fare ciò, partendo dal presupposto che il formato di output dell'applicazione è un dispositivo tablet, si è reso necessario un componente che semplificasse la gestione di questi spostamenti dei giocatori usando il touch del tablet. E TouchScript¹ rappresenta lo strumento migliore al caso, per la gestione delle gestures touch.

5.2.1 TouchScript per Unity3D

TouchScript è una libreria per Unity3D per il riconoscimento delle gestures in Unity.

Le principali caratteristiche di TouchScript sono le seguenti:

- supporto ad una svariata serie di dispositivi, a partire dagli schermi degli smartphone (iOS, Android, Windows Store/Windows Phone) per arrivare agli schermi touch di grandi dimensioni;

¹<http://touchscript.github.io/>

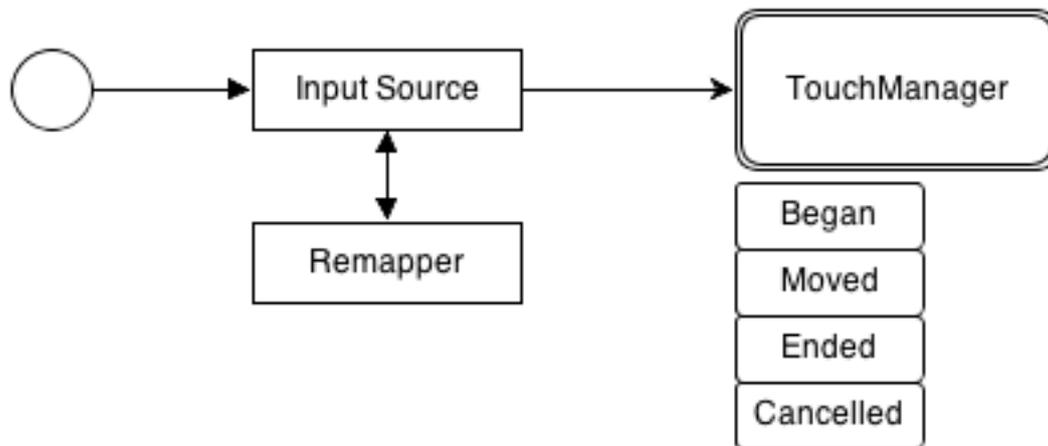


Figura 5.2: Schema che raffigura come viene riconosciuta una gesture dalla libreria TouchScript.

Fonte: <https://github.com/TouchScript/>

- riconoscimento delle comuni gesture: pressione, rilascio, tap, pressione a lungo, pinch, scalamento, rotazione;
- gestione delle gesture in modo tale da rispettare la gerarchia delle gesture impostata per far sì che sia rispettato l'ordine;
- documentazione approfondita e svariato numero di esempi per rendere semplice l'utilizzo;
- possibilità di simulare il multi-touch su un dispositivo no-multi-touch (attivandolo con Alt+click);
- licenza open-source gratuita.

Nel caso di questo progetto di tesi, per rendere i giocatori trascinabili nel campo grazie al trascinamento touch, è stato inserito su ogni modello dei giocatori il componente *Transform Gesture*. Questo componente riconosce la gesture della pressione, il conseguente spostamento e rilascio e calcola la nuova posizione del giocatore che sarà visualizzato nella nuova posizione (e nelle posizioni intermedie durante lo spostamento). Il punto forte di questo componente è che semplifica il richiamo a specifiche callback che possono essere chiamate in determinati momenti della gesture, che possono essere discrete o continue.

Le gesture discrete sono illustrate di seguito.

- Tap Gesture — riconosce un singolo, doppio o triplo tap;
- Press Gesture — riconosce quando un oggetto è stato premuto;
- Release Gesture – riconosce quando è terminata la pressione di un oggetto;
- Long Press Gesture – riconosce una pressione a lungo su un oggetto;
- Flick ² Gesture – riconosce uno swipe rapido su un oggetto.

²La gesture *Flick* è definita come un normale swipe però fatto in maniera più rapida: <https://brianthurston.com/swipe-vs-flick-on-ios-6ff6a516b6aa>

Le gesture discrete sono:

- Transform Gesture — riconosce una trasformazione come traslazione, rotazione, scalamento o la loro combinazione.
- Screen Transform Gesture — simile alla precedente ma in coordinate dello schermo
- Pinned Transform Gesture – simile alla Transform Gesture ma se un oggetto è stato fissato nel suo centro, potrà solo ruotare e scalare attorno al suo centro senza muoversi.
- Meta Gesture – rimappa tutti gli eventi touch per un oggetto a cui è collegato come eventi separati.

Nel caso di questa applicazione ha rivestito importanza fondamentale la gesture *Transform Gesture* e gli eventi da essa scaturiti. *TransformStarted*, *Transformed*, *TransformCompleted* sono i tre eventi ai quali si possono iscrivere determinate callback per chiamare metodi realizzati ad-hoc.

In particolare è stato sfruttato l'evento *TransformStarted* per gestire il disegno delle linee sul campo, in base ai movimenti dei giocatori (come descritto in sezione 5.4, seguendo la simbologia ufficiale FIBA ³). Quando inizia la trasformazione - provando a muovere un giocatore - viene eseguito un metodo che provvede a duplicare il giocatore trascinato per creare un effetto ghost, in modo tale da indicare posizione di partenza e posizione di arrivo, come è mostrato nella Figura 5.3. Quando la trasformazione è in corso è chiamato il metodo iscritto all'evento *Transformed* che si occupa di aggiornare il componente *LineRenderer* che disegna le linee in base ai dati costantemente aggiornati durante la trasformazione. Per quanto riguarda invece l'evento *TransformCompleted* è necessario specificare che il suo funzionamento è importante perché è grazie ai metodi iscritti a questo evento che è possibile memorizzare nella timeline alcune informazioni fondamentali come gli eventi di passaggio e blocco. Più precisamente: nel momento in cui la palla - tra due diversi frame - è stata spostata da un giocatore all'altro, viene rilevato un passaggio grazie al metodo *DetectBall* del componente *Ball.cs* aggiunto all'oggetto della palla. Questo metodo viene lanciato quando la palla viene rilasciata nella posizione del giocatore a cui si vuole passare la palla e oltre ad uno snap al giocatore ricevitore (la palla viene ancorata al ricevitore), viene creato nella timeline un Evento di tipo passaggio, utile alla successiva riproduzione tridimensionale dell'azione della corretta animazione.

5.3 Moduli fondamentali dell'applicazione

In questa sezione sono descritti i moduli fondamentali, tutti scritti in C# e ideati per funzionare su Unity come *Component* da associare ad un *GameObject* per estenderne le sue funzionalità e rispettare i requisiti alla base della progettazione.

5.3.1 Gestione giocatori

Questo modulo, che è rappresentato dal componente *PlayersManager*, si occupa della gestione dei giocatori in campo. In particolare contiene due strutture dati

³Federazione Internazionale di Pallacanestro: <http://www.fiba.basketball/>

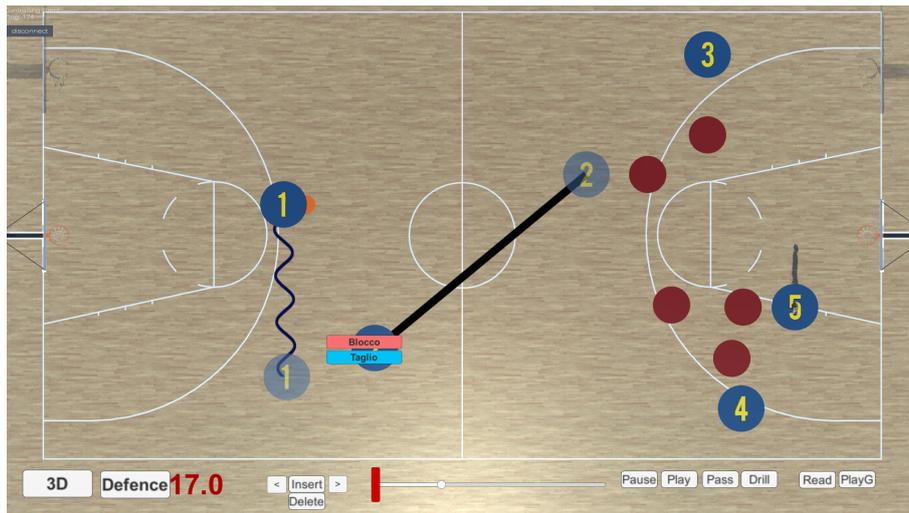


Figura 5.3: Esempio di generazione delle frecce in base al trascinarsi dei giocatori sul campo

che contengono le liste dei giocatori che corrispondono alle due squadre di attacco e difesa. Questo componente si rivela di grande importanza perché con esso è possibile in qualsiasi momento accedere ai riferimenti dei giocatori presenti in campo tramite i metodi *GetHomePlayersList* e *GetAwayPlayersList*, rispettivamente i giocatori della squadra che attacca e quelli della squadra che difende. Con esse dunque si può ottenere qualsiasi dato corrispondente al giocatore tramite i componenti che sono aggiunti al *GameObject* del giocatore preso in considerazione. Il componente della gestione dei giocatori è richiamato spesso durante il funzionamento del programma e uno delle sue implementazioni fondamentali è in stretta connessione con il componente che gestisce l'inserimento dei keyframe nella timeline temporale. Nel momento in cui l'utente che sta creando una tattica clicca sul tasto *Insert* il componente *AnimationManager* fa riferimento al componente *PlayersManager* per ottenere le informazioni riguardo le posizioni dei giocatori in quel determinato momento per memorizzarle nella timeline. La Figura 5.4 è illustrato il funzionamento all'interno del programma dell'inserimento del keyframe.

5.3.2 Generazione percorso

La generazione del path, ossia il percorso che il giocatore deve percorrere una volta che è stato premuto il tasto *Play*, è assegnata al componente *PathCreator*. Questo componente si occupa di creare i percorsi dei giocatori e far partire i loro movimenti e per fare ciò è stato utilizzato un pacchetto scaricato dall'Asset Store di Unity che ha il nome di *Simple Waypoint System* (SWS) ⁴.

Quest'ultimo si presenta come una estensione che permette di creare waypoint e path in maniera semplice attraverso un editor visuale o tramite codice. Una volta creati questi path (insieme di waypoint) si può facilmente assegnare a un oggetto e far partire il movimento di esso attraverso il path a lui assegnato. SWS fornisce uno strumento per creare path e interpolare attraverso essi il movimento degli oggetti e i

⁴<https://www.rebound-games.com/products/sws/>

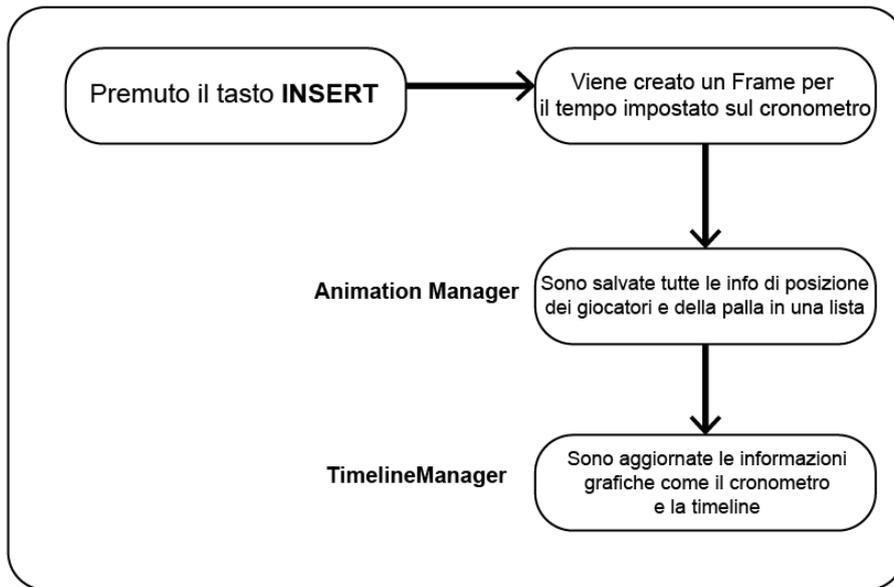


Figura 5.4: Diagramma del funzionamento dell'inserimento di un keyframe.

tempi di questo movimento possono essere calcolati utilizzando uno svariato numero di fattori per personalizzare l'interpolazione.

Le principali caratteristiche di SWS sono:

- editor per la creazione e gestione dei path;
- possibilità di creare path lineari, curvi, utilizzando curve di Bézier e le Nav-Mesh di Unity;
- script già inclusi per gestione del movimento;
- funziona con la modalità 2D o 3D di Unity;
- esempi già presenti per rendere più facile il primo utilizzo;
- codice scritto interamente in C#.

Nel momento in cui si inserisce un keyframe, viene creato un componente di tipo *Transform*, un oggetto che contiene informazioni spaziali, per ogni giocatore con le informazioni in quel momento temporale. Questi oggetti, che si definiscono *Waypoints*, sono imparentati al componente *PathCreator* che è l'incaricato a gestire questo tipo di oggetti per la creazione dei path. Quando l'allenatore preme il tasto *Play* il *PathManager* crea a partire dai waypoints, creati ad ogni inserimento di keyframe, i percorsi (path) per ogni giocatore e lo assegna ad ognuno di essi. In Figura 5.5 si possono vedere i percorsi in colore violetto creati dal *PathCreator*.

Una volta aggiunto il path all'oggetto del giocatore, viene azionato il movimento e il giocatore comincia a muoversi lungo il percorso in base ai tempi memorizzati tra un keyframe e l'altro. Per far sì che il giocatore si muova tra un keyframe e l'altro utilizzando la corretta velocità che rispetti la differenza di tempo tra un keyframe e l'altro, è stato modificato il componente *splineMove*, nativo di SWS.

Più in particolare il sistema di SWS permette di azionare un qualsiasi tipo di evento, ogni qual volta un determinato oggetto incontra un waypoint. E questo dà



Figura 5.5: Sul campo sono disegnati i path (colore viola chiaro) e i waypoint (colore giallo) creati alla pressione del tasto Play.

la possibilità di iscrivere agli eventi scatenati una determinata callback e dunque lanciare un metodo in questo momento. Ciò permette di chiamare nel momento in cui un oggetto, che nel caso di questo progetto è il giocatore, entra in contatto con un waypoint (questo è dovuto perché a entrambi gli oggetti è assegnato un *Collider* che al contatto scatena la callback che chiama un metodo assegnato) viene chiamato da *PathCreator* il metodo *SetSpeed* che calcola la velocità con cui il giocatore deve percorrere quel tratto di percorso in base alla distanza tra il punto di arrivo e quello di partenza e il tempo che intercorre tra di loro. Questo valore di velocità calcolato viene passato al componente *splineMove* che deve essere assegnato ad ogni giocatore per far sì che si muova. La Figura 5.6 rende più chiaro di come viene gestito e azionato il movimento dei giocatori con un diagramma a blocchi che mostra il funzionamento della creazione dei percorsi a partire dalla pressione del pulsante Play.

Il componente *PathCreator* gestisce, in maniera simile a come viene fatto con il cambio di velocità, anche l'azionamento delle animazioni che i giocatori devono effettuare quando il programma viene utilizzato vestendo un visore di realtà virtuale e dunque i giocatori devono essere visualizzati in tre dimensioni. Nel paragrafo successivo si descrive più approfonditamente questo aspetto.

5.3.3 Generazione contenuti 3D dalle coordinate 2D

Oltre a visualizzare il movimento dei giocatori in 2D, il lavoro di tesi ha tra le sue funzioni quella di proporre la visualizzazione del movimento in tre dimensioni in modo tale che i giocatori che vestono il visore di realtà virtuale possono fruire dell'esperienza dal loro punto di vista e vedere i giocatori che si muovono nel campo.

Per fare ciò anche in questo caso il componente *PathCreator* descritto nel paragrafo precedente svolge un ruolo fondamentale. Il suo funzionamento consiste nel gestire i valori booleani dell'*Animator*, il componente assegnato ad ogni oggetto che gestisce le animazioni.

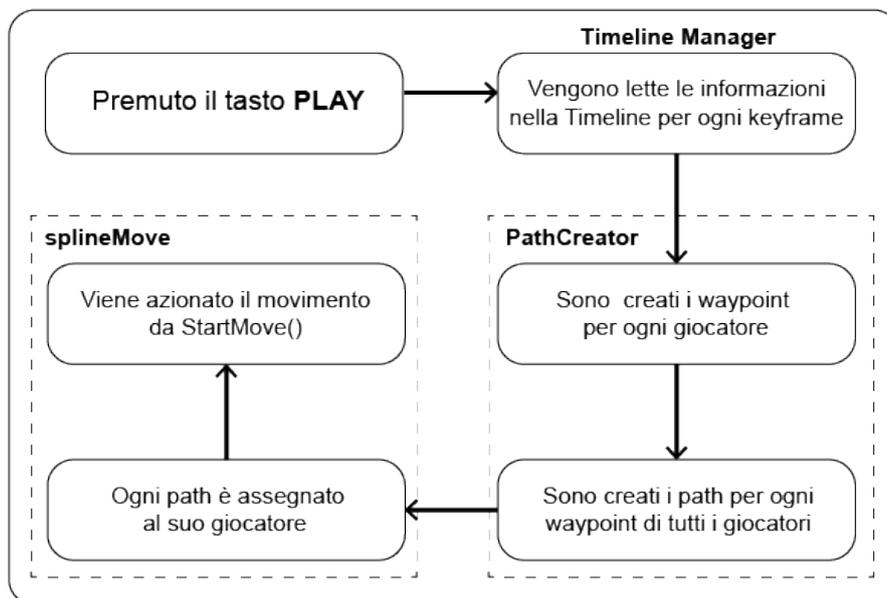


Figura 5.6: Workflow di creazione dei percorsi attraverso i quali si muovono i giocatori

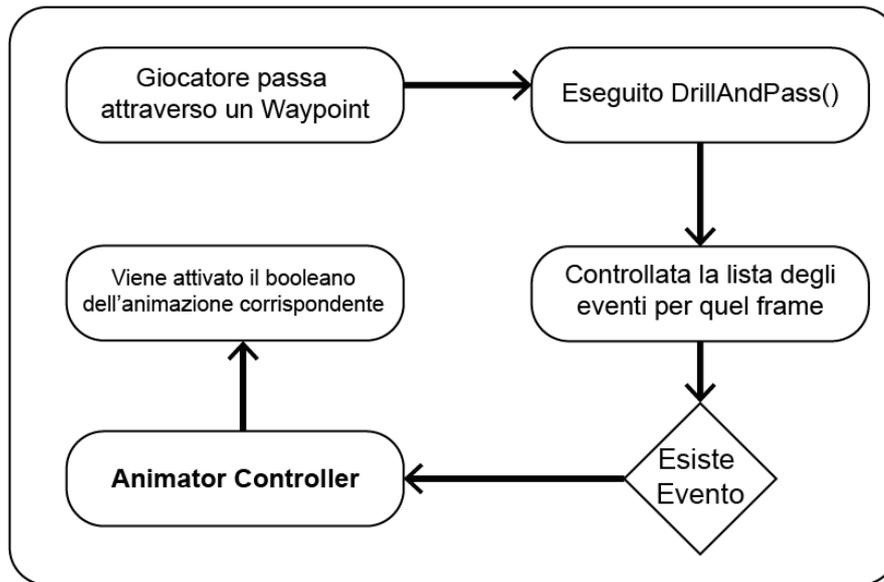


Figura 5.7: Schema che mostra il funzionamento base del metodo DrillAndPass

Il componente che gestisce le animazioni: Animator Controller

Il componente Animator ⁵ è usato per assegnare le animazioni ai GameObject nella scena. Esso richiede un riferimento all' *Animator Controller* che definisce le animazioni da usare e controlla quando e come gestire le transizioni tra di loro. L'Animator Controller viene creato all'interno degli asset di Unity e consente di organizzare e gestire un set di animazioni per i personaggi dei giocatori. Questo tipo di approccio permette di avere più animazioni gestite con un unico strumento e passare da una all'altra quando si verificano determinate condizioni di gioco.

Il controller ha riferimenti alle clip di animazione utilizzate al suo interno e gestisce i vari stati di animazione e le transizioni tra di loro utilizzando una cosiddetta macchina a stati, che si può considerare come una sorta di diagramma di flusso o un semplice programma scritto in un linguaggio di programmazione visuale all'interno di Unity.

Queste condizioni di gioco sono dettate dai valori che il PathManager genera in base ai valori impostati dall'allenatore durante la creazione della tattica o dai valori del machine learning nel caso in cui si scelga la modalità di fruizione della tattica caricandola dai dati di STATS SportVU. Il metodo *DrillAndPass* che viene chiamato ad ogni waypoint per ogni giocatore, ha il compito di gestire le animazioni di palleggio e passaggio dei giocatori. Il funzionamento, descritto nella Figura 5.7, si basa sul leggere ad ogni keyframe la lista degli eventi memorizzata nella struttura dati. Questa lista degli eventi è creata o in base ai dati inseriti dall'allenatore o a quelli che sono importati utilizzando il file delle coordinate di STATS SportVU. Dunque, quando è letto un evento dalla lista, esso provoca il cambiamento dei valori booleani dell'Animator del giocatore interessato da quell'evento e questo permette ad esso di muoversi correttamente.

Di seguito sono elencate le animazioni inserite in questo progetto di tesi e i valori booleani che le scatenano e nella Figura 5.9 è mostrata una panoramica che permette di cogliere al volo le differenti animazioni e i vari collegamenti tra di loro

⁵<https://docs.unity3d.com/Manual/class-Animator.html>

che permettono le transizioni tra ognuna di esse. Nella Figura 5.8 si vedono le impostazioni di transizione da un'animazione all'altra: le condizioni booleane che la scatenano e il tempo di transizione. Esso può essere modificato per decidere quanto tempo deve passare per portare la transizione alla nuova animazione. Ogni freccia nel diagramma a blocchi corrisponde alla transizione da una animazione all'altra.

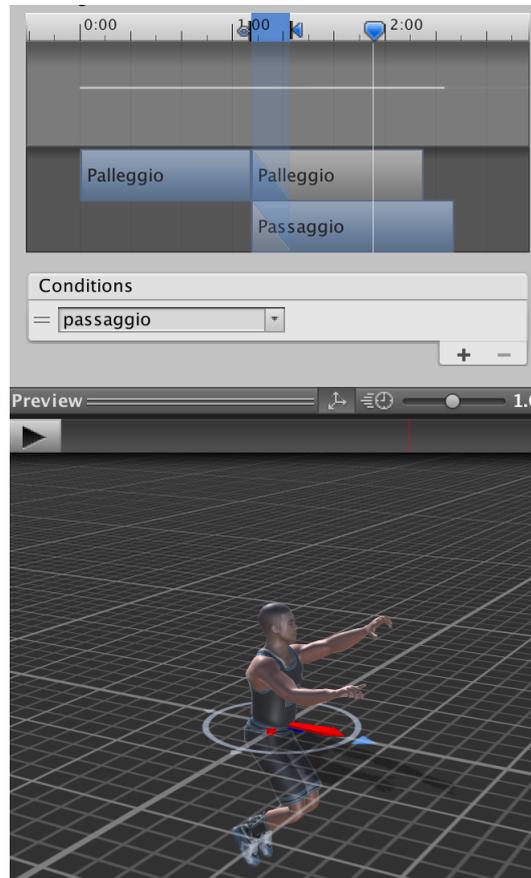


Figura 5.8: Schermata di configurazione della transizione da una animazione all'altra. La barra temporale permette di gestire il passaggio tra le due e *Conditions* consente di aggiungere i valori booleani

Su ogni freccia si possono e devono stabilire le condizioni per transire tra le varie animazioni e queste condizioni, che possono essere più di una, è possibile modificarle via codice ed è questo quello che è stato fatto usando la combinazione del PathCreator e del componente *MoveAnimator*. Quest'ultimo è assegnato ad ogni giocatore ed è quello attraverso il quale passano i valori booleani da trasferire all'Animator per le animazioni.

A causa della somiglianza nella struttura ossea, è possibile mappare le animazioni da uno scheletro umanoide a un altro, consentendo retargeting e cinematica inversa. Con rare eccezioni, ci si può aspettare che i modelli umanoidi abbiano la stessa struttura di base, rappresentando le parti articolate principali del corpo, della testa e degli arti. Il sistema Mecanim fa buon uso di questa idea per rendere più facile il rigging e il controllo delle animazioni. Un passo fondamentale nella creazione di un'animazione consiste nell'impostare una mappatura tra la struttura ossea umanoide semplificata di Mecanim e le ossa reali dello scheletro; nella terminologia Mecanim, questa mappatura è chiamata Avatar. Nella figura Figura 5.10 si vede

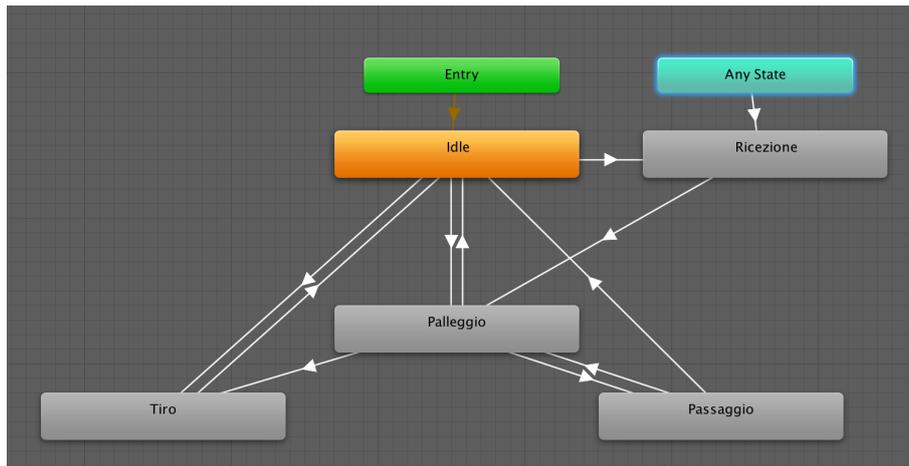


Figura 5.9: L'Animator assegnato ad ogni giocatore con le animazioni di palleggio, passaggio, tiro, ricezione interconnesse tra di loro dalle condizioni scatenanti.

lo scheletro del Mecanim di Unity al quale sono assegnate le ossa dello scheletro del modello importato.

5.3.4 Gestione del tempo

Le informazioni appartenenti ai giocatori in campo in ogni determinato keyframe sono memorizzate nella timeline ogni volta che ne viene aggiunto uno. Lo schema in Figura 5.11 raffigura la struttura dati progettata per il funzionamento del programma di questo elaborato di tesi. Ogni oggetto di tipo *Timeline*, che può essere considerata come l'insieme delle informazioni dell'azione creata, è composta da *Frame* che si possono pensare come i keyframe veri e propri i quali contengono le informazioni dei giocatori (e la palla) riferite solo al quel determinato istante di tempo.

Ogni oggetto di tipo Frame contiene:

- informazioni riguardo l'istante di tempo di quel frame;
- lista dei giocatori di attacco (posizione dei giocatori);
- lista dei giocatori di difesa (posizione dei giocatori);
- informazioni di posizione della palla;
- nome del giocatore che è in possesso della palla a quel keyframe.

Il contenuto delle timeline è fondamentale al fine della riproduzione dell'animazione perché il metodo PathCreator elabora le informazioni contenute in essa per creare i waypoint che sono le basi per costruire i percorsi sui quali far muovere i giocatori.

5.3.5 Networking

Il lavoro di tesi proposto permette ai giocatori di seguire le indicazioni tattiche dell'allenatore in una sessione in diretta in cui l'allenatore dirige lo spostamento dei

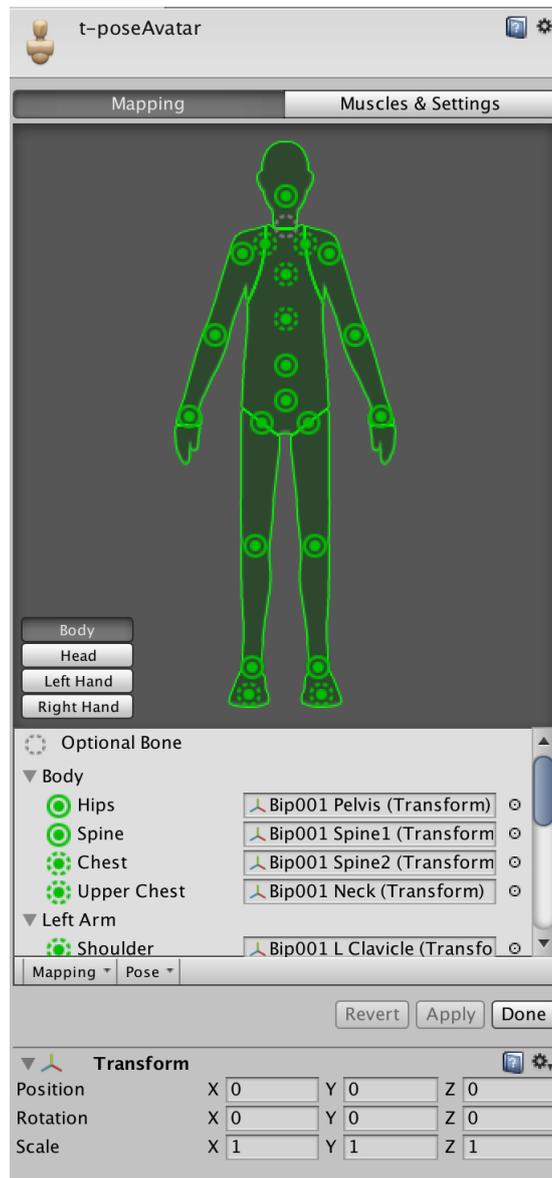


Figura 5.10: Questa è la configurazione in cui al Mecanim di Unity si assegnano le ossa del modello creato in Maya.



Figura 5.11: La struttura di ogni oggetto di tipo Timeline.

giocatori che tramite un visore in realtà virtuale seguono in tempo reale i movimenti. Per realizzare ciò è stato utilizzato PUN Networking, che è un motore di rete indipendente e piattaforma di networking che è stato scelto a discapito del motore di rete integrato in Unity (Unity Networking), perché dispone di una documentazione di gran lunga più approfondita e questo ha permesso una facilità di sviluppo maggiore.

Di seguito sono elencate le principali differenze tra PUN e Unity Networking.

1. host model: Unity Networking è basato su client server. I server vengono eseguiti tramite un client Unity (quindi tramite uno dei giocatori). Photon è basato su server-client ugualmente, ma ha un server dedicato; Questo evita che le connessioni siano interrotte a causa della disconnessione degli host;
2. connettività: Unity Networking funziona con NAT punch-through per cercare di migliorare la connettività: poiché i giocatori ospitano i server di rete, la connessione spesso fallisce a causa di firewall / router ecc. La connettività dunque non può mai essere garantita e c'è un basso tasso di successo. Photon ha un server dedicato, non c'è bisogno di NAT punch-through o altri concetti simili. La connettività è garantita al 100 % e nel caso raro di connessione fallita, questo è dovuto a una rete lato client molto severa (ad esempio una VPN aziendale);
3. performance: poiché i server Unity sono hostati dagli utenti il ping sarà sempre più alto di quello dei server dedicati di Photon e soprattutto non fa affidamento sulla connessione del giocatore che sta attuando come server;
4. prezzo: il prezzo di PUN è gratuito come Unity Networking e con la licenza gratuita possono collegarsi in contemporanea fino a 100 utenti. In questo caso non ci saranno mai, tenendo conto di tutta la squadra più lo staff, più di 20 persone collegate contemporaneamente, il che rende PUN più che efficace;
5. documentazione e supporto: Unity non sembra dare molta priorità alla loro implementazione di Networking, soprattutto offrendo una scarsa documentazione. Raramente ci sono miglioramenti alle funzionalità e le correzioni ai bug sono rarissime. Photon viene aggiornato costantemente e alcune parti sono disponibili con il codice sorgente, a differenza di Unity Networking di cui non si può accedere al sorgente.

L'applicazione progettata grazie a questo elaborato di tesi ha tra le sue funzionalità quella che fornisce ai giocatori uno strumento per visualizzare in tempo reale i loro spostamenti sul campo decisi dall'allenatore che li controlla attraverso la sua applicazione sul proprio tablet.

Per questo motivo il progetto si compone di una applicazione server dedicata agli allenatori, in cui è presente la maggior parte dei blocchi che servono al funzionamento dell'applicazione: creazione tattica su una timeline, salvataggio, caricamento di una già salvata, play, pausa e switch alla visualizzazione 3D; l'altra parte costituente si tratta dell'applicazione che è utilizzata dai giocatori che si compone di un semplice modulo che si collega in tempo reale al server e riceve i dati creati dall'applicazione dell'allenatore e di conseguenza si comporta come un visualizzatore di tattiche create dal coach. Il funzionamento di PUN si basa su tre script principali:

- Photon Network Manager: questo è il componente principale di PUN e si occupa di stabilire la connessione e gestire la connessione tra l'applicazione dell'allenatore e il visualizzatore dei giocatori. Il suo funzionamento consiste nell'attivarsi una volta che è aperta l'applicazione dell'allenatore, creando un server al quale, una volta aperta la loro applicazione, i giocatori si connettono in modo automatico, portando a un livello di estrema semplicità la connessione tra i dispositivi, perché non si richiede alcuna configurazione particolare di rete. Nella Figura 5.12 si può notare il funzionamento di questo componente che mostra le informazioni sulla connessione;

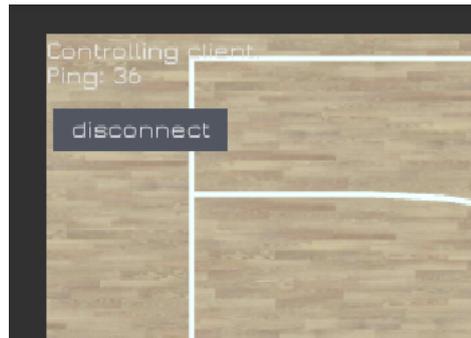


Figura 5.12: Grafica che informa la situazione attuale di rete e permette di iniziare o terminare il collegamento. La dicitura “Controlling Client” significa che si sta utilizzando l’applicazione server, mentre quella client se scritto “Receiving Updates”

- Photon View: è il corrispondente script NetworkView di Unity Networking. Ogni oggetto (prefab) di cui si vuole tenere traccia dei movimenti tra server e client deve necessariamente avere Photon View tra i suoi componenti, altrimenti i movimenti effettuati su server non avranno alcun effetto su client. Per fare questo agli oggetti dei giocatori nella scena è stato aggiunto questo componente a cui, oltre ad un ViewId - che serve per distinguere gli oggetti tra di loro - è stata assegnata la referenza del componente Transform, per far sì che Photon View “osservi” le variazioni di posizione dell’oggetto del giocatore e le trasferisca attraverso la rete ai client attraverso lo script Photon Network Manager.
- Photon Animation View: questo componente è assegnato ai modelli 3D dei giocatori e serve a trasmettere attraverso la rete i trigger booleani dell’Animator⁶. Più precisamente il funzionamento si basa in questo modo: nell’applicazione server sono presenti i modelli 3D dei giocatori le cui animazioni sono scaturite dai dati inseriti dall’allenatore. Questi dati generano le animazioni base di palleggio, passaggio, blocco, taglio e tiro che sono azionate nel momento in cui vengono cambiati dei valori booleani in base alle informazioni inserite dall’allenatore. Per far sì che anche nei client collegati le animazioni siano riprodotte in maniera corretta, è necessario inserire in ogni modello 3D del giocatore questo componente, che si occupa di inviare lo stesso valore booleano che viene inserito nella macchina a stati del giocatore su server anche a quello del client. Il componente, come si vede in Figura 5.14, permette di

⁶l’Animator è un componente di un prefab di Unity che si occupa di gestire le animazioni dei personaggi tramite una macchina a stati

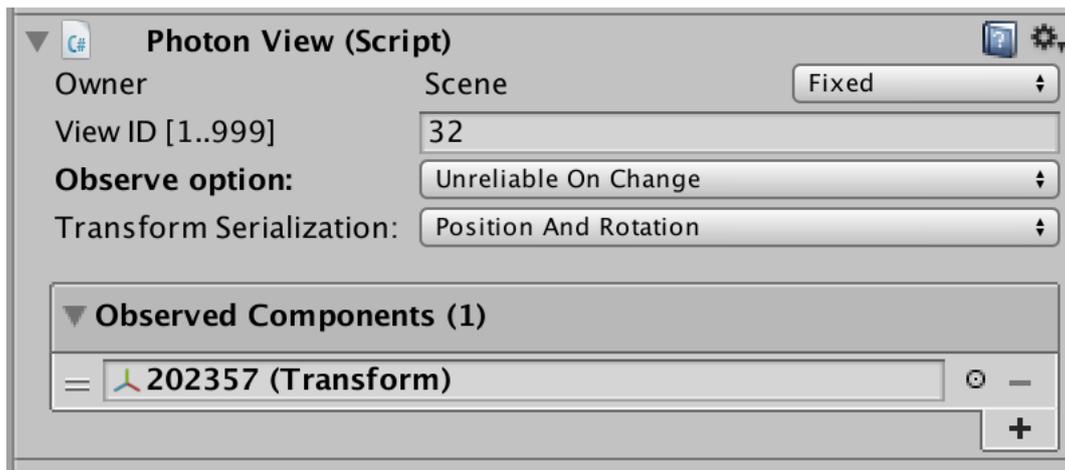


Figura 5.13: Il componente PhotonView con il ViewID scelto e i componenti da tener traccia tra i quali compare il Transform dell'oggetto di cui si vogliono trasmettere i movimenti attraverso la rete

scegliere di quali valori booleani trasmettere i valori e dunque in questo modo quali animazioni riprodurre sia su server che su client. In questo caso è stato abilitato ogni valore booleano delle animazioni create ad-hoc per il movimento dei giocatori di pallacanestro.

5.4 Utilizzo dell'applicazione

In questa sezione si elencano le funzionalità principali del programma che è stato progettato secondo i requisiti discussi nel paragrafo (4.1).

Le funzioni del programma realizzato sono le seguenti:

- possibilità di creare tattiche
- possibilità di salvare tattiche, caricarle e gestirle
- riproduzione delle tattiche in 2D
- riproduzione delle tattiche in 3D
- riproduzione in realtà virtuale real-time delle tattiche sui dispositivi dei giocatori collegati all'allenatore
- caricamento di porzioni di partita precedentemente registrate con il sistema di tracking dei giocatori di NBA STATS SportVU
- possibilità di riprodurre quest'ultime in 2D / 3D / realtà virtuale
- possibilità di modificare le tattiche importate da STATS SportVU e mostrare ai giocatori soluzioni correttive

Il funzionamento principale di questo progetto si basa sull'utilizzo in contemporanea di due applicazioni simili, una per l'allenatore e una per i giocatori che sono collegati in contemporanea all'allenatore mentre egli comanda i movimenti dei giocatori. Di seguito si spiega l'utilizzo prima della applicazione per gli allenatori e poi quella dei giocatori.

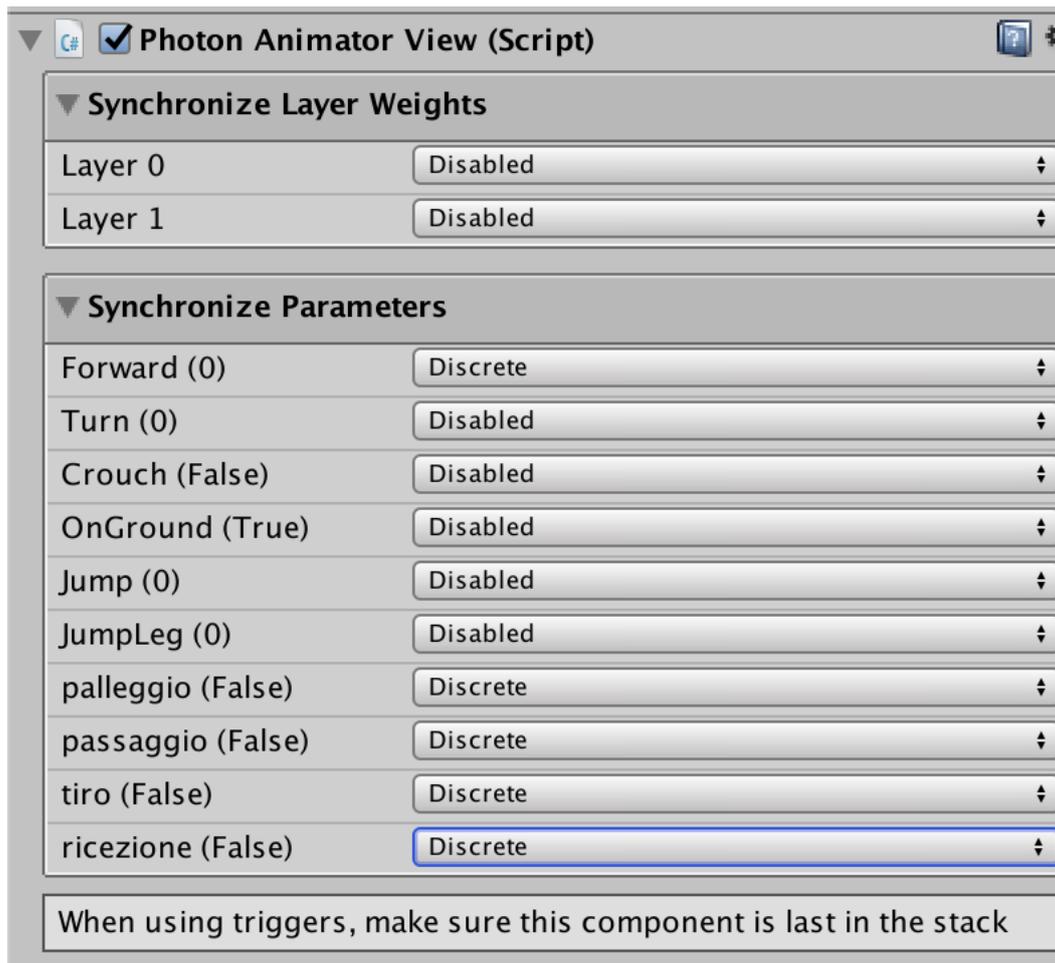


Figura 5.14: Il componente Photon Animation View e i valori booleani scelti da trasmettere

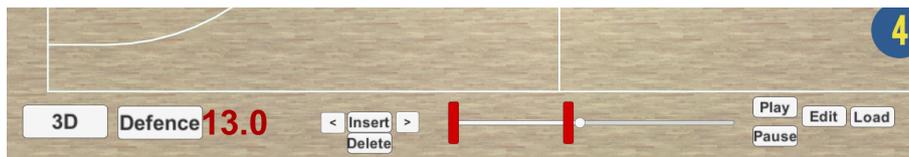


Figura 5.15: La barra degli strumenti del programma di creazione di tattiche.

5.4.1 Programma per allenatori

Layout principale

La schermata con cui l'allenatore interagisce dopo aver avviato il programma consiste principalmente in due parti fondamentali. Nella parte centrale ed estesa a gran parte del display è mostrato il campo, con i giocatori già piazzati nella posizione di partenza di default.

Nella parte inferiore dello schermo è presente una serie di tasti che servono all'allenatore a gestire i giocatori in campo e ad inserire i keyframe nella timeline, oppure a poter caricare una timeline già salvata in precedenza o un file di STATS SportVU. Visibili nella Figura 5.15, più precisamente i tasti sono illustrati di seguito.

- con il tasto *Defence* si può abilitare o disabilitare la visualizzazione della difesa e questo permette all'allenatore di preparare le sue tattiche scegliendo o meno di far vedere ai giocatori i movimenti in campo dei difensori;
- con il tasto *Insert* si memorizzano le posizioni attuali dei giocatori a quel tempo e si memorizzano nella timeline;
- con il tasto *Delete* è possibile invece cancellare un keyframe precedentemente creato;
- il tasto *Edit* ha lo scopo di attivare la Edit Mode per gli allenatori che consiste nel apportare modifiche allo schema caricato. Durante questa modifica e ogni qualvolta l'allenatore sta muovendo i giocatori sullo schermo, gli utenti collegati nella sessione live non vedranno gli spostamenti dal loro punto di vista ma da quello della tribuna dell'impianto sportivo. Una volta terminato l'edit, si ritorna alla visione in prima persona;
- le frecce servono a spostarsi nel tempo sulla timeline e vedere le informazioni per ogni keyframe.
- i tasti *Save* e *Load* servono a gestire il salvataggio e il caricamento delle azioni memorizzate nella timeline non solo a partire da un'azione creata in precedenza dall'allenatore, ma anche dai dati di coordinate che sono forniti da STATS SportVU. Il tasto *New* permette di creare una nuova timeline da zero;
- i tasti *Play* e *Pause* sono i tasti che gestiscono la riproduzione dell'animazione;
- poco più a sinistra dei tasti *Insert* e *Delete* c'è il *Cronometro* che indica in ogni momento qual è il tempo dell'azione in corso.

Workflow di utilizzo

Una volta deciso se creare lo schema con la difesa o meno, l'allenatore può, sfruttando il touchscreen del suo display, muovere i giocatori sul campo a suo piacimento. Nel momento in cui un giocatore viene mosso, vengono visualizzate sul display i movimenti dei giocatori in questo modo: del movimento effettuato dal giocatore sono indicati due momenti, quello di partenza e quello di arrivo. Il primo è visualizzato attraverso il modello del giocatore con la texture trasparente, mentre il punto di arrivo è indicato dalla stessa mesh ⁷ ma con texture opaca, come si può vedere nella figura Figura 5.16. Inoltre, una freccia è disegnata tra le due mesh e questa indica la direzione di movimento. Dato che il giocatore può effettuare diversi tipi di movimento con e senza palla, sono state individuate delle modalità di interazioni che prendendo spunto dalla simbologia ufficiale riconosciuta universalmente hanno definito dei paradigmi di utilizzo di questo programma.

Più nello specifico, dato che il programma è stato ideato per la realizzazione su tablet si è deciso una modalità di interazione simile a quella dei programmi presenti sul mercato descritti nel (Capitolo 2), che dunque presentasse l'utilizzo delle frecce della simbologia di disegno della pallacanestro, ma che fosse adattata all'utilizzo su un dispositivo tablet. La principale differenza sta nel fatto che in questo programma gli allenatori spostano i giocatori e automaticamente si crea la freccia che ne determina il movimento, mentre nei programmi già esistenti l'allenatore deve inserire appositamente la freccia per ottenere il movimento del giocatore e questo rappresenta un passaggio macchinoso che può essere bypassato con la soluzione proposta.

In questo caso invece l'allenatore muove il giocatore che genera una freccia nella direzione di andamento e questa freccia è automaticamente disegnata in base al tipo di movimento. Ad esempio se il giocatore ha il possesso della palla (questo lo si nota perché la palla si muove insieme al giocatore, se quest'ultimo viene mosso), la linea che è disegnata sarà ondulata, secondo le norme di disegno FIBA. Per quanto riguarda invece i movimenti senza palla ci sono due tipi di freccia che si possono generare e questo dipende dalla scelta che viene effettuata nel momento in cui viene rilasciato il giocatore (in questo caso senza palla) e viene chiesto se il movimento è un *taglio* o un *blocco*. Per quanto riguarda il passaggio, invece, basta spostare la palla da un giocatore all'altro e verrà creata una linea tratteggiata che ne indica il passaggio dal passatore al ricevitore. Nella Figura 5.17 sono mostrati i due tasti che sono descritti di seguito.

1. *Blocco*: questo tasto crea la linea classica del blocco (a forma di T) e genera all'interno della timeline l'evento di tipo Blocco che richiamerà l'animazione correlata.
2. *Taglio*: questo tasto crea la linea classica del taglio (una freccia che ha la punta nella direzione in cui si muove il giocatore).

Una volta mossi i giocatori in campo e selezionato quanto tempo deve durare questo movimento, spostandosi del tempo necessario sulla timeline, si può memorizzare questa disposizione in campo per il frame in questione cliccando sul tasto Insert: un marker di colore rosso segnala l'inserimento del keyframe sulla timeline,

⁷Con *mesh* si indica il modello 3D poligonale dell'oggetto

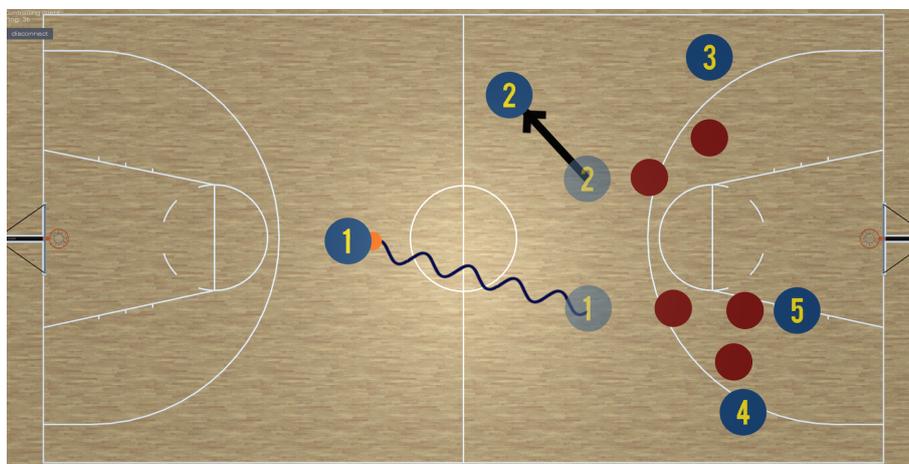


Figura 5.16: Qui è mostrato un keyframe appena inserito: i giocatori sono mostrati nel loro punto di partenza (mesh trasparente) e arrivo (mesh opaca) con una freccia che descrive il loro movimento

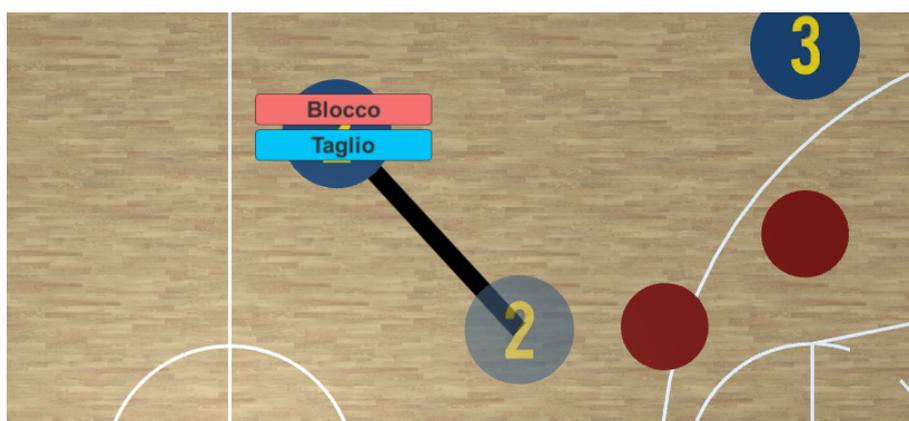


Figura 5.17: La possibilità di scegliere se il movimento è un blocco o un taglio è permessa da questo pulsante di scelta

che si trova nella parte centrale in basso. In questo modo i dati sono memorizzati e si può passare a lavorare sul keyframe successivo perché le mesh trasparenti della posizione di partenza del giocatore al keyframe precedente spariscono; la posizione di arrivo del keyframe precedente ora rappresenta la posizione di partenza del keyframe attuale. Muovendo i giocatori si creano allo stesso modo le frecce appena descritte. Dal keyframe successivo il funzionamento è lo stesso e, avanzando nella timeline, l'utente può creare un'azione completa. Se si vuole tornare indietro per visionare i movimenti dei keyframe precedenti o per modificarli, lo si può fare senza problemi cliccando sulle frecce per muoversi nel tempo: il cronometro visualizza il secondo dell'azione. Verranno visualizzati i keyframe memorizzati con le frecce che indicano il movimento verso quello successivo. Se si ha bisogno di eliminare il keyframe, si clicca sul tasto *Delete* che cancella il keyframe.

Quando l'azione è completa ci sono varie opzioni: essa può essere salvata cliccando sul tasto *Save* e può essere riprodotta cliccando sul tasto *Play*. Cliccando sul tasto *Load* si può invece:

1. caricare una tattica precedentemente creata e poi salvata dall'allenatore;
2. caricare una porzione di partita a partire dai dati forniti da NBA STATS SportVU.

Questi dati sono raccolti durante le partite di NBA e corrispondono a ogni movimento fatto dai giocatori e dalla palla e si presentano in un file di testo con le coordinate X, Y e Z che sono poi elaborate in Unity per trasformarle nello stesso formato di quelli che, inseriti dall'allenatore durante la creazione manuale della tattica, vanno a riempire la *Timeline*.

Allo stesso modo con cui un allenatore può riprodurre una tattica appena creata, si può fare con la porzione di partita importata cliccando sul tasto *Play*. L'animazione parte e i giocatori si muovono seguendo le coordinate del file di STATS che rappresentano i movimenti che il giocatore ha realmente fatto sul campo. Per quanto riguarda gli eventi di passaggio, tiro e blocco, essi in questo tipo di modalità non sono inseriti manualmente dall'allenatore perché si tratta di una parte di partita importata dal file di statistiche NBA. Un altro lavoro di tesi [29] permette di elaborare questo tipo di informazione mancante ed è usato in questo caso per fornire tutti gli elementi per far partire le animazioni di passaggio (in Figura 5.18), di tiro (in Figura 5.19) e passaggio (in Figura 5.20) anche nella modalità di fruizione attraverso le statistiche NBA.

A questo punto l'allenatore, oltre alla mera riproduzione di questo file, può apportare le azioni correttive, muovendosi all'interno della timeline e spostare i giocatori, perché ha bisogno di suggerire ai suoi giocatori un movimento diverso da quello che è stato analizzato dalla partita reale. Dopo aver apportato le modifiche, si può far partire la riproduzione dell'azione nelle stesse modalità descritte in precedenza.

Queste sono le funzionalità principali del programma che può utilizzare l'allenatore. Ovviamente uno dei punti di forza di questo progetto è quello che consente di creare una sessione in cui i giocatori possono seguire in diretta l'allenatore dare loro indicazioni riguardo le tattiche. Questo è possibile grazie al modulo di networking descritto nella sezione (5.3.5) che permette ai giocatori di seguire quanto indicato dall'allenatore attraverso un visore di realtà virtuale. Nel seguito è spiegato il funzionamento base della versione del programma da utilizzare con un visore.



Figura 5.18: Un frame tratto dall'animazione del passaggio.



Figura 5.19: Un frame tratto dall'animazione di tiro.



Figura 5.20: Un frame tratto dall'animazione del passaggio.



Figura 5.21: Visualizzazione dell'esperienza immersiva percepita dal punto di vista del giocatore.

5.4.2 Programma per i giocatori

Il funzionamento del programma per i giocatori deve prevedere una semplicità di interazione massima perché deve permettere un rapido collegamento del giocatore con la sessione in diretta dell'allenatore e per questo motivo si limita soltanto all'utilizzo del *gaze* per scegliere al principio del programma il giocatore che si deve impersonare nell'esperienza di realtà virtuale. Questa scelta è possibile attraverso il menù iniziale che visualizza 5 cubi con i numeri dei 5 giocatori in campo. Posizionandosi con il punto di vista verso il cubo del giocatore che si vuole impersonare, basta fissarlo per un determinato numero di secondi e verrà caricata la scena della palestra con i giocatori in campo.

La scena 3D si presenta così all'utente che ha scelto il giocatore da impersonare: la telecamera virtuale è imparentata alla testa del giocatore scelto all'inizio e questo permette di vivere l'azione dal suo punto di vista come è possibile vedere nella Figura 5.21.

Quando un allenatore muove i giocatori sul campo durante la gestione delle tattiche i giocatori, onde evitare un fastidio percettivo dovuto al proprio avatar che viene trascinato nello spazio, vedono lo scenario dall'inquadratura posta nella tribuna dell'impianto sportivo come è possibile osservare nella Figura 5.22. Questo consente loro di vedere la tattica durante la fase di creazione e nel momento in cui il coach preme il tasto Play, la telecamera ritorna ad essere posizionata sulla testa del giocatore impersonato.

Di seguito, nella Figura 5.24 è presente uno schema riassuntivo dell'utilizzo parallelo delle due applicazioni (quella per l'allenatore e quella per il giocatore) e ciò che viene visualizzato contemporaneamente nelle rispettive interfacce.

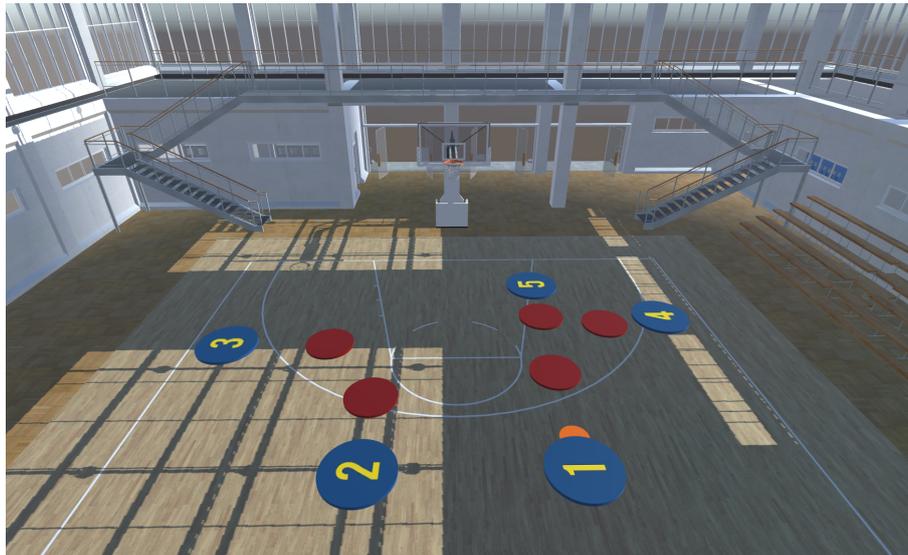


Figura 5.22: La visualizzazione che viene proposta nel visore di realtà virtuale quando l'allenatore sta modificando le posizioni dei giocatori in campo.

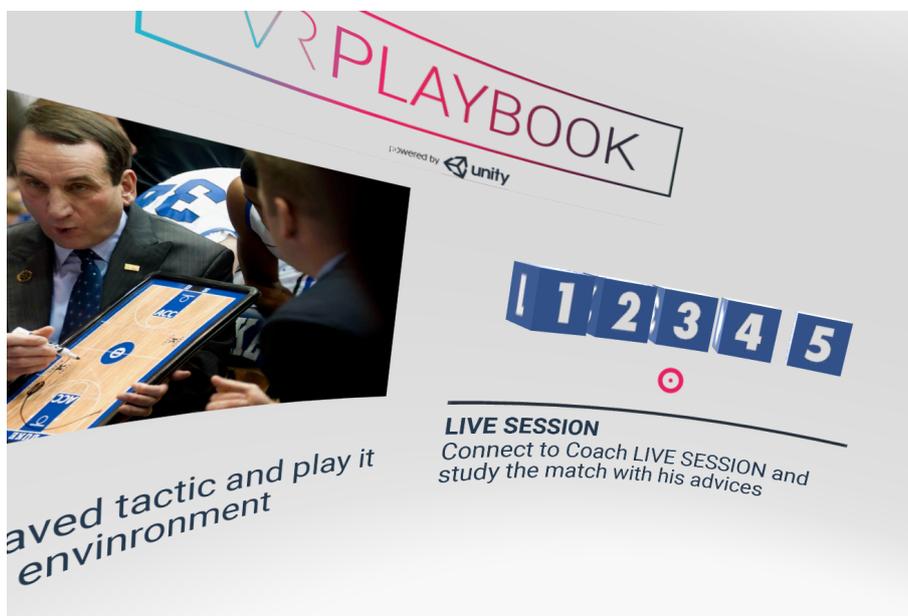


Figura 5.23: La schermata di inizio dell'esperienza VR dove si può scegliere il giocatore da impersonare nello scenario tridimensionale

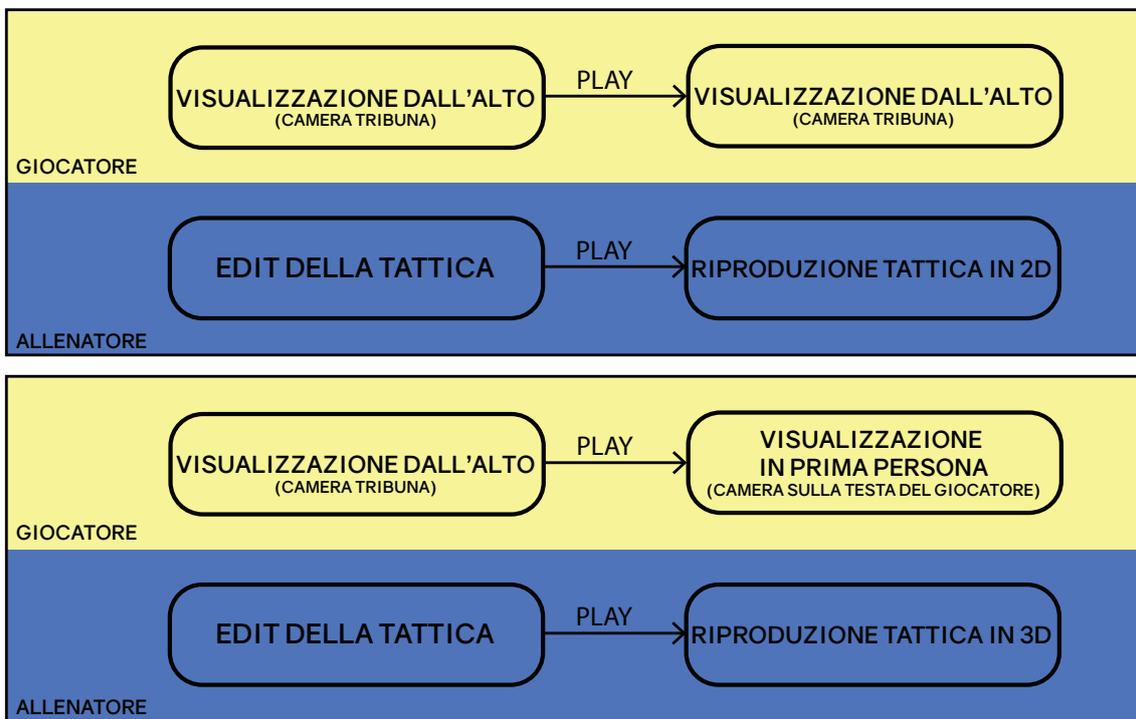


Figura 5.24: Un riassunto di cosa succede contemporaneamente nelle due applicazioni per allenatore e giocatori.

Capitolo 6

Conclusioni

In questo capitolo conclusivo vengono inserite alcune considerazioni sul lavoro svolto, evidenziando i punti di forza e quelli di debolezza, nonché i possibili sviluppi futuri o le attività da svolgere per migliorare i risultati.

6.1 Considerazioni finali

L'elaborato proposto propone un nuovo strumento per gli allenatori di pallacanestro per la creazione, la gestione e la loro riproduzione in realtà virtuale, con l'obiettivo di aumentare le performance di training dello schema. L'utente crea le proprie tattiche attraverso un sistema touch, in cui può muovere i giocatori a suo piacimento e memorizzare le posizioni in una timeline temporale che raccoglie le informazioni di tutti i giocatori. Queste tattiche sono poi si possono riprodurre in 2D, 3D, ma soprattutto in realtà virtuale, che è l'obiettivo principale di questo lavoro. La fruizione avviene attraverso un visore di realtà virtuale, come ad esempio il Samsung Gear VR, che viene indossato dal giocatore che in tempo reale osserva le indicazioni del coach che comanda tutto dalla sua applicazione server. L'allenatore può creare le tattiche da mostrare ai giocatori oppure caricare quelle già salvate, ma anche importare nel sistema una porzione di partita formata dai dati di STATS SportVU, un sistema che registra i movimenti dei giocatori di tutte le partite di NBA e mette a disposizione le coordinate nel formato X, Y, Z.

Per la realizzazione del progetto sono stati prima discussi i requisiti con l'allenatore Francesco Raho, capo del settore giovanile dell'Auxilium Pallacanestro Torino. Una volta fissati i requisiti ed esplorato lo stato dell'arte degli strumenti già presenti, si è deciso di elaborare questa applicazione. L'applicazione è stata creata in Unity, utilizzando Visual Studio Code per scrivere i moduli in C# per rispettare i requisiti stabiliti in partenza.

Il sistema creato ha rispettato i requisiti decisi con Raho, seppur potrebbero esserci migliorie future, che dipendono dallo sviluppo della tecnologia e da implementazioni dell'applicazioni, che sono discusse nella parte finale di questo capitolo.

6.2 Possibili implementazioni e lavori futuri

Il programma proposto da questo elaborato di testi propone uno scenario tridimensionale immersivo in un contesto di pallacanestro, con l'obiettivo di migliorare l'apprendimento cestistico. Dunque, il miglioramento futuro di questa applicazione si lega fortemente sul progresso di questi strumenti alla base. Certamente una miglioria nella qualità dei display dei visori di realtà virtuale porterebbe ad una esperienza sempre più immersiva e percepita reale. Inoltre, a parte le ovvie possibili migliorie al comparto grafico e al movimento dei giocatori che può essere migliorato in maniera esponenziale facendo indossare la tuta ai giocatori e registrare i loro propri movimenti, porterebbe l'esperienza su un livello più alto la possibilità di scaricare direttamente gli schemi da servizi come Synergy (di cui si è parlato ad esempio nello stato dell'arte al Capitolo 2) che offrono una galassia incredibilmente vasta di soluzioni tattiche già salvate. Sarebbe inoltre un vantaggio poter integrare il sistema di STATS per avere già a portata di mano tutte le partite nel formato di coordinate X, Y, Z di cui si serve questo programma.

Queste ultime funzioni appena citate arricchirebbero di gran lunga la versione di questo elaborato usato dagli allenatori. Per quanto riguarda invece l'applicazione che usano i giocatori sarebbe uno sviluppo importante quello di aumentare l'interattività tra allenatore e giocatore al fine di creare un dialogo costruttivo con l'obiettivo del miglioramento prestazionale. Ad esempio si potrebbe ideare l'introduzione di un oggetto usato dal giocatore per superare la visualizzazione passiva dello schema e portarla verso una dimensione interattiva in cui il giocatore può controllare alcune parti dell'esperienza.

Di seguito sono descritte le possibili interazioni che potrebbero essere introdotte dall'utilizzo di un dispositivo di input (può essere un joystick o un leap motion, un mouse o qualsiasi altro dispositivo simile) utilizzato dai giocatori.

- possibilità per il giocatore di relazionarsi con lo staff, l'allenatore e i giocatori. Più precisamente si potrebbe creare una possibile interazione che permetterebbe al giocatore di chiedere assistenza durante la visualizzazione dello schema, inviando così all'allenatore le informazioni temporali e spaziali del movimento che il giocatore ha bisogno di approfondire in maniera più dettagliata. Allo stesso modo il giocatore potrebbe avere la possibilità di suggerire all'allenatore le sue considerazioni e questo aumenterebbe l'interattività tra i vari componenti della squadra e di conseguenza la possibilità di raggiungere un miglioramento maggiore;
- possibilità di usare questo dispositivo di input per fruire di una sessione offline in single-mode, cosa che al momento invece è limitata all'obbligo della esperienza multi-utente in diretta con l'allenatore che è il server. In questo modo l'utente potrebbe rivedersi le sessioni già provate in diretta (o registrate appositamente dall'allenatore per la fruizione a casa), muovendosi nel tempo e nello spazio, ad esempio avendo l'opportunità di spostarsi da un giocatore all'altro, studiando così i movimenti dei suoi compagni;
- aggiunta di una sezione training che sia ludica ma allo stesso tempo educativa. L'idea sarebbe quella di trasporre il lavoro similmente fatto al rugby, descritto nel Capitolo 2 nel contesto di educazione della capacità decision-making nel

basket. L'idea è quella di creare uno scenario in cui ci sono azioni create appositamente dall'allenatore per fermarsi in un punto ben preciso. In questo momento verrebbe posto il quesito ai giocatori su come si interverrebbe nel proseguimento dell'azione. Una volta effettuata la scelta, il giocatore vedrebbe il risultato della sua decisione e nel caso in cui fosse quella sbagliata, gli verrebbe mostrato quello che sarebbe successo se avesse scelto in modo corretto. E su questa base si potrebbe pensare di costruire un sistema a punti per stimolare la gamification di questo processo di apprendimento nelle squadre di basket, soprattutto quelle dei settori giovanili, in cui il fattore ludico potrebbe rappresentare un punto di forza per l'insegnamento di concetti tattici che ai giocatori più giovani potrebbe risultare difficile con le tradizionali modalità.

Bibliografia

- [1] William R Sherman e Alan B Craig. *Understanding virtual reality: Interface, application, and design*. Elsevier, 2002.
- [2] D. Katz et al. *Calibration of virtual reality systems*. US Patent App. 14/589,755. Lug. 2015. URL: <https://www.google.com/patents/US20150193983>.
- [3] S. LaValle e P. Giokaris. *Perception based predictive tracking for head mounted displays*. US Patent App. 14/285,470. Dic. 2014. URL: <https://www.google.com/patents/US20140354515>.
- [4] Steven M. Lavallo. *Virtual reality*. 2015.
- [5] John R Stowers et al. «Virtual reality for freely moving animals». In: *Nature Methods* 14.10 (ott. 2017), pp. 995–1002. ISSN: 1548-7091. DOI: 10.1038/nmeth.4399. URL: <http://https://doi.org/10.1038/nmeth.4399>.
- [6] Douglas Nitz. «A place for motion in mapping». In: *Nature neuroscience* 18.1 (2015), pp. 6–7.
- [7] Mel Slater e Sylvia Wilbur. «A framework for immersive virtual environments (FIVE): Speculations on the role of presence in virtual environments». In: *Presence: Teleoperators and virtual environments* 6.6 (1997), pp. 603–616.
- [8] ISPR. *International Society for Presence Research. (2000). The Concept of Presence: Explication Statement*. 2000. URL: <https://ispr.info/about-presence-2/about-presence/>.
- [9] Richard L Gregory. *Eye and brain: The psychology of seeing*. Princeton university press, 2015.
- [10] *Sensorama simulator*. US Patent 3,050,870. Ago. 1962.
- [11] Ivan E Sutherland. «The ultimate display». In: *Multimedia: From Wagner to virtual reality* (1965).
- [12] Ivan E Sutherland. «A head-mounted three dimensional display». In: *Proceedings of the December 9-11, 1968, fall joint computer conference, part I*. ACM. 1968, pp. 757–764.
- [13] S. S. Fisher et al. «Virtual Environment Display System». In: *Proceedings of the 1986 Workshop on Interactive 3D Graphics. I3D '86*. Chapel Hill, North Carolina, USA: ACM, 1987, pp. 77–87. ISBN: 0-89791-228-4. DOI: 10.1145/319120.319127. URL: <http://doi.acm.org/10.1145/319120.319127>.
- [14] URL: <https://www.theverge.com/2015/3/1/8127445/htc-vive-valve-vr-headset>.
- [15] Karl M Kapp. *The gamification of learning and instruction: game-based methods and strategies for training and education*. John Wiley & Sons, 2012.

- [16] David L Porter. *Basketball: A Biographical Dictionary*. Greenwood Publishing Group, 2005.
- [17] FIP. *Modello di allenamento integrato: la costruzione degli esercizi*. 2011. URL: <http://www.fip.it/public/22/1073/replay-2-0-capobianco.pdf>.
- [18] Michael J Tarr e William H Warren. «Virtual reality in behavioral neuroscience and beyond». In: *nature neuroscience* 5 (2002), pp. 1089–1092.
- [19] Benoit Bideau et al. «Using virtual reality to analyze sports performance». In: *IEEE Computer Graphics and Applications* 30.2 (2010), pp. 14–21.
- [20] Benoit Bideau et al. «Using virtual reality to analyze links between handball thrower kinematics and goalkeeper’s reactions». In: *Neuroscience letters* 372.1 (2004), pp. 119–122.
- [21] Benoit Bideau et al. «Real handball goalkeeper vs. virtual handball thrower». In: *Presence: teleoperators and virtual environments* 12.4 (2003), pp. 411–421.
- [22] Sébastien Brault et al. «Virtual kicker vs. real goalkeeper in soccer: a way to explore goalkeeper’s performance». In: *Movement & Sport Sciences* 3 (2015), pp. 79–88.
- [23] Jean-Philippe Petit, Hubert Ripoll et al. «Scene perception and decision making in sport simulation: a masked priming investigation.» In: *international Journal of sport psychology* 39.1 (2008), pp. 1–19.
- [24] Yohann Cardin et al. «Investigate Naturalistic Decision-Making of Football Players in Virtual Environment: Influence of Viewpoints in Recognition». In: *NDM11, the 11th International Conference on Naturalistic Decision Making*. 2013, pp. 109–117.
- [25] Nicolas Vignais et al. «Which technology to investigate visual perception in sport: Video vs. virtual reality». In: *Human movement science* 39 (2015), pp. 12–26.
- [26] Yazhou Huang, Lloyd Churches e Brendan Reilly. «A Case Study on Virtual Reality American Football Training». In: *Proceedings of the 2015 Virtual Reality International Conference*. VRIC ’15. Laval, France: ACM, 2015, 6:1–6:5. ISBN: 978-1-4503-3313-9. DOI: 10.1145/2806173.2806178. URL: <http://doi.acm.org/10.1145/2806173.2806178>.
- [27] Joe Willage. «Using VR to improve free throw percentage in the NBA». In: ().
- [28] Alexandra Covaci et al. «A virtual reality simulator for basketball free-throw skills development». In: *Technological Innovation for Value Creation* (2012), pp. 105–112.
- [29] Gianpaolo Basilicò. *Generazione Automatica di Contenuti 3D per Applicazioni di Realtà Virtuale mediante tecniche di Machine Learning*. 2017.
- [30] STATS LLC. *STATS SportVU*. 2017. URL: <https://www.stats.com/sportvu-basketball/>.
- [31] Unity Technologies. *Unity User Manual (2017.1)*. 2017. URL: <https://docs.unity3d.com/Manual/GameObjects.html>.