

POLITECNICO DI TORINO

III Facoltà di Ingegneria

Corso di Ingegneria delle Telecomunicazioni

Tesi di Laurea Magistrale

**Applicazione di un visore olografico
a uno Spatial Decision Support System
per operazioni sul territorio mediante UAV**



Relatore:

Poessore Andrea Bottino

Candidato:

Fabio Antoci

Luglio 2018

Indice

1 Introduzione	3
2 Stato dell'arte	4
2.1 tecnologie di riferimento	5
2.1.1 Spatial Decision Support System	5
2.1.2 SMAT	7
2.1.3 Virtual, Augmented e Mixed Reality	8
2.1.4 Intel "commercial drone solutions for enterprise"	11
2.2 Valutazioni economiche	12
3 Requisiti del progetto	14
3.1 Obiettivi progetto	14
4 Progettazione e Sviluppo	15
4.1 Vista di contesto	15
4.1.1 Utilizzatori della applicazione	15
4.1.2 Dati in ingresso	16
4.1.3 Dati Geografici di contesto:	16
4.2 Vista funzionale	19
Figura 4.3 vista funzionale	19
4.2.1 Visualizzazione dinamica dei terreni	19
4.2.2 Visualizzazione Layer tematici	20
4.2.3 Visualizzazione dei voli UAV	21
4.2.4 Impostazione di una ricognizione UAV	22
Figura 4.6 esempio progettazione volo	22
4.2.4.1 Traiettoria	22
4.2.4.2 Impostazioni Field Of View	24
4.3 Vista tecnologica	24
4.3.1 Server locale	25
4.3.2 Servizio WMS	27
4.3.3 Applicazione UWP per Hololens	27
4.4 Aspetti realizzativi	29
4.4.1 Terreni	29
4.4.2 Sistemi di riferimento	34
4.4.3 UX/UI	39
5 Valutazioni	47
5.2 Sviluppi futuri	51
Bibliografia	53

1 Introduzione

Negli ultimi anni si è visto un incremento esponenziale dell'utilizzo di Unmanned Aircraft System (UAS). Grazie all'abbattimento dei costi, queste tecnologie si sono potute diffondere in diversi ambiti professionali e commerciali. Da una veloce analisi economica si può vedere come il campo di maggior investimento sia quello della raccolta dati e analisi dei terreni. Questo, più altri fenomeni qua non citati, hanno portato ad un aumento dei dati, specialmente tridimensionali, di oggetti fisici. Nasce così il concetto di Digital Twin, ovvero una raccolta di dati massiva, cumulativa, real-time, real-world su molteplici dimensioni. Queste misurazioni possono tenere traccia, e creare, modelli di sviluppo degli oggetti interessati.

Un esempio di Digital Twin applicato ai territori lo si può trovare in uno studio recentemente pubblicato dalla NASA "*3D View of Amazon Forests Captures Effects of El Niño Drought*"^[1]. In questo studio la NASA ha creato un modello 3D estremamente accurato della foresta Amazzonica mediante delle ricognizioni con tecnologia LiDAR (300,000 pulsazioni al secondo) non possibili da terra o dallo spazio. Ripetendo questo procedimento per tre anni di fila hanno potuto analizzare i danni reali, in termini di alberi caduti e rinvigorimento della foresta, causati dalla siccità, nominata *El Niño*, in confronto con gli anni "normali".

Nonostante il continuo sviluppo sull'elaborazione e analisi dei dati tridimensionali la fruizione di questi rimane principalmente 2D, mediante schermi piatti, fatta eccezione di alcune applicazioni in realtà virtuale. Quest'ultima rimane dipendente da una infrastruttura circostante (computer, sensori, controlli) e isola l'utente che si vede costretto a scegliere tra un mondo totalmente virtuale o il mondo reale.

Nel 2015 compare il primo device in ambito commerciale per la Mixed Reality, prodotto da Microsoft sotto il nome di HoloLens.

L'HoloLens sta avendo un notevole successo, ancorché diffuso solo per la ricerca è sviluppo, molte aziende stanno adattando i loro software o creandone di nuovi per poter fornire nuovi servizi mediante questo device. Gli ambiti applicativi dove c'è più fermento sono quello del Training e del Remote Support. Oltre a questi due settori la Mixed Reality e Augmented Reality spaziano in molteplici settori industriali, cavalcando anche l'onda della Industry 4.0, ovvero quella che viene definita quarta rivoluzione industriale portata dall'introduzione di nuove tecnologie (principalmente IoT) in campo industriale.

Tra queste l'applicazione della Boeing^[2], da cui prende spunto questo lavoro. Per la prima volta viene utilizzato un visore olografico per dare supporto alla decisioni in una situazione critica territoriale di incendio boschivo. Dando una visione tridimensionale del territorio interessato e delle squadre in azione si è riscontrata una più veloce e migliore interpretazione della situazione da parte di chi coordinava le azioni.

Da qui parte questo lavoro di tesi, cercando di valutare e sperimentare le possibilità della Mixed Reality in ambito di decisioni territoriali. Vedendo la Augmented Reality come veicolo per interpretare e trasmettere a terzi dati territoriali in maniera più efficiente. E consentire, mediante la Mixed Reality, di configurare ed adattare lo spazio circostante per avere il miglior environment di lavoro e, con esso, la User Experience

2 Stato dell'arte

Lo stato attuale delle applicazioni sviluppate per HoloLens, si colloca in un contesto totalmente sperimentale, dato che ad oggi non sono ancora presenti applicazioni di tipo commerciale. Tra queste applicazioni sperimentali risaltano quelle in collaborazione con la stessa Microsoft. Non essendo mirate alla commercializzazione spesso hanno poca documentazione associata.

È questo il caso dell'applicazione di Mission Control per incendi boschivi della Boeing sviluppata in collaborazione con Microsoft, con l'obiettivo di portare il software INEXIA, sviluppato da INSITU, da una visione 2D Top-Down ad una accurata visualizzazione tridimensionale (Figure 2.1). Di questa applicazione esiste solo un video ufficiale che ne mostra i propositi ed un primo funzionamento e qualche articolo specialistico.



Figura 2.1: Immagine dell'applicazione della Boeing presa dal video di presentazione

Si dichiara che sia la prima applicazione ad utilizzare la Mixed Reality per la gestione di unmanned aerial systems (UAS) al fine di aiutare il contenimento di incendi. L'operatore può manipolare e controllare le mappe tattiche olografiche. Tenendo sotto controllo la situazione nel suo insieme può inoltre decidere di inviare un unmanned aerial vehicle (UAV) semplicemente trascinandolo nella scena. Da quel momento in poi il veicolo saprà da solo cosa fare. L'obiettivo vuole essere dare al capo missione una visione più chiara e tridimensionale di ciò che accade in modo da poter prendere decisioni e dare indicazione alle varie squadre più velocemente e in maniera più informata.

2.1 tecnologie di riferimento

2.1.1 Spatial Decision Support System

I Spatial Decision Support System (SDSS), servono a fornire una serie di strumenti per facilitare decisioni riguardanti problemi legati, direttamente o indirettamente, alla spazialità. SDSS sono progettati per essere intuitivi, suggerire soluzioni attraverso la presentazione di diverse alternative, per essere flessibili ed adattarsi facilmente a diversi scenari e per supportare metodi analitici ed euristici di analisi. Questo non implica che questi sistemi generino direttamente una decisione finale, Piuttosto, con l'utilizzo dei SDSS, viene semplificata e resa più comprensibile la decisione all'utente, lavorando attivamente con i dati tramite opportuni modelli rappresentativi, Per raggiungere questi obiettivi solitamente un SDSS include un database, dei spatial model, delle interfacce, la possibilità di visualizzare dati e generare report.

I SDSS sono l'evoluzione dei Decision Support System (DSS) integrati con i Geographic Information System (GIS). I DSS sono una classe di sistemi informativi dedicati al supporto al processo decisionale. All'utente finale, lavorando attivamente con i dati tramite modelli rappresentativi, viene semplificata e resa più comprensibile la decisione. Il framework generico di un DSS consiste nell'integrazione di tre elementi (Figura 2.2). Il primo è un gestore di dati, il secondo un gestore dei modelli ed infine un interfaccia. Solitamente la gestione dei dati viene demandata ad un database che contiene tutti dati che possono essere utilizzati nel sistema. Il gestore di modelli si occupa di trattare tutte le funzionalità modelli analitici, alimentati a loro volta dai dati del database. L'interfaccia si occupa di fornire un'interazione tra l'utente finale ed il sistema dando la possibilità di interagire con i modelli e di visualizzare l'output di quest'ultimi.

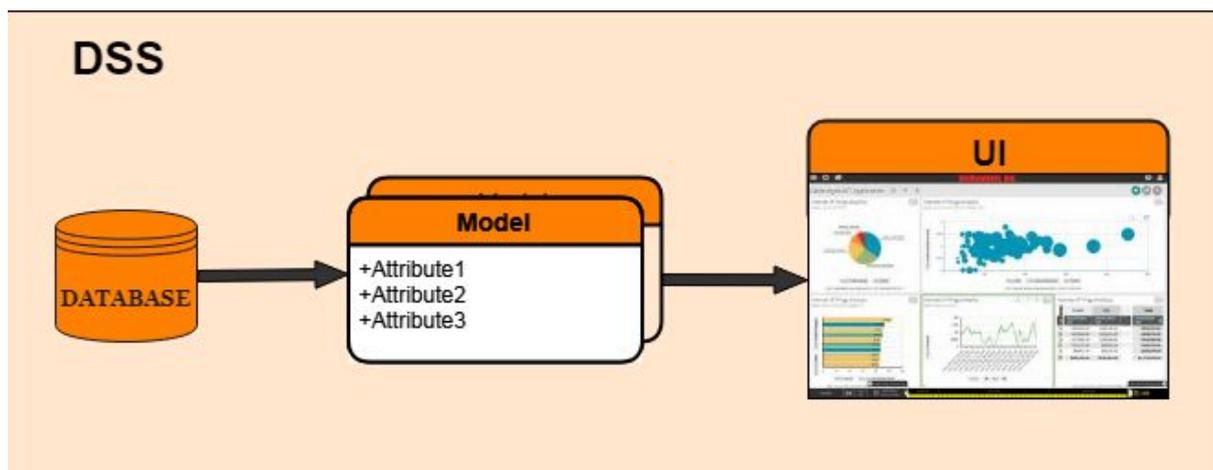


Figure 2.2: Schema dei tre componenti dei DSS

L'idea dei DSS nasce nei primi anni '70 con il lavoro di Gorry e Scott-Morton (1971)^[3] dove definiscono un DSS come un sistema interattivo che aiuti a risolvere problemi non strutturati, o semi-strutturati, attraverso l'utilizzo di dati e modelli. Negli anni '80 Alter (1980)^[4] produce il primo esempio di DSS e ne espande la struttura, portandola ai tre componenti fondamentali illustrati sopra.

Quasi contemporaneamente si sviluppano i Geographic Information System o GIS. Il primo utilizzo su grande scala avviene negli anni '60 con il Canadian Land Inventory Project che cercò di determinare i differenti usi delle varie aree sul territorio e i possibili usi futuri di diverse aree^[5]. Nonostante ciò nei primi decenni lo sviluppo di questi sistemi fu molto limitato a causa delle difficoltà nella portabilità dei software e dei dati, degli alti costi di mantenimento e la dalla difficoltà di utilizzo tramite command-line. Inoltre la prima generazione di GIS forniva delle capacità di modellazione troppo ridotte per poter essere usata per decisioni commerciali. L'evoluzione tecnologica dell'hardware dei computer e, conseguentemente, delle capacità di calcolo, negli anni 1980 1990, portò ad uno sviluppo dei GIS anche a livello commerciale. Nel 1982 la ESRI (Environmental Systems Research Institute) rilasciò il primo software GIS commerciale chiamato ArcInfo.

La seconda e più significativa evoluzione dei GIS avviene nella seconda metà degli anni '90 con la diffusione massiva delle tecniche di rilevazione aerea e satellitare nel settore civile e la successiva esplosione del Web che ha reso accessibili questi dati a livello consumer. In questo periodo numerose aziende come ArcIMS, GeoMedia e ESRI rilasciano servizi di Web Mapping. Affianco a queste realtà commerciali nascono anche realtà open-source (es. University of Minnesota MapServer).

Nonostante la costante evoluzioni dei GIS, la mancanza di modelli analitici e l'impossibilità di presentare tecniche di valutazione di scenari, non li consente di essere usati come sistemi di supporto alle decisioni. Da qui nascono i SDSS come unione dei DSS e GIS (Figura 2.3).

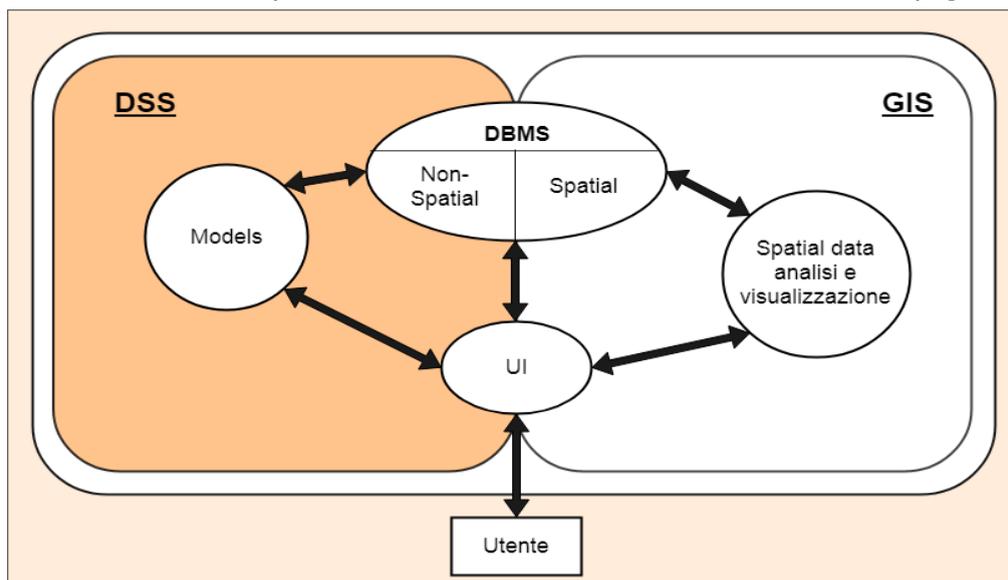


Figura 2.3: schema SDSS, unione tra DSS e GIS

I modelli analitici (Models), tipici dei DSS, forniscono svariate elaborazioni dei dati acquisiti dal database (DBMS) basati su calcoli statistici, matematici o valutazioni su strutture a criteri multipli. Allo stesso tempo, i GIS forniscono spatial data, funzioni di analisi di questi dati e possibilità di visualizzazione su mappe. Entrambi i sistemi si interfacciano all'utente tramite la user interface (UI) arricchendola con le rispettive informazioni. Si visualizzano dunque dati geo-referenziati, come mappe territoriali, ma su questi si potranno anche avere informazioni derivate dai modelli analitici, in questo modo il sistema fornisce un valido supporto per decisioni e pianificazioni in ambito territoriale.

Un esempio di utilizzo è quello della valutazione dei rischi e la pianificazione di contromisure durante un incendio.(Figura 2.4).

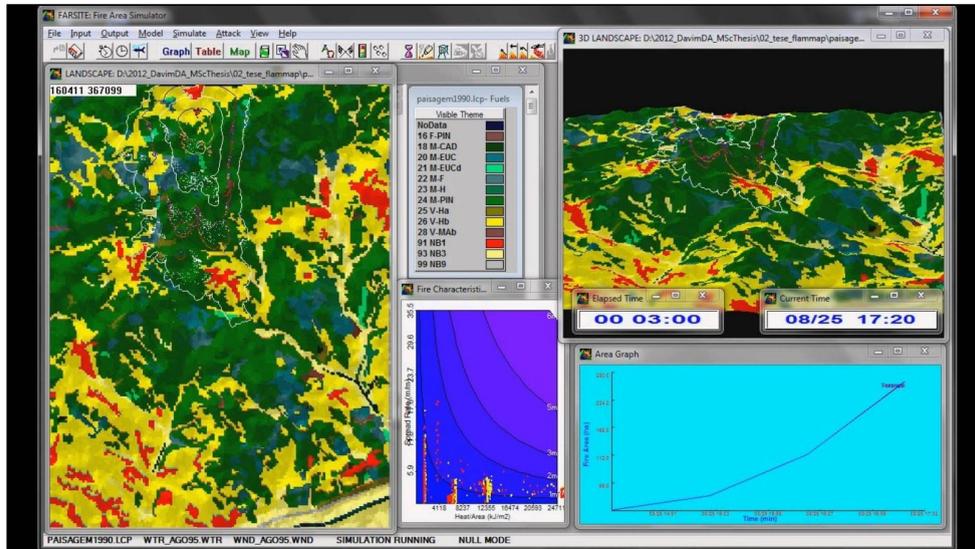


Figura 2.4: esempio di SDSS per la simulazione delle aree interessate da incendi

Spesso la filosofia dei SDSS viene integrata direttamente nei software di pianificazione e/o gestione di missione. Permettendo in alcuni casi di effettuare delle simulazioni digitali, rendendo ancora più intuitivo il risultato delle analisi all'utente finale e consentendo anche a chi non ha conoscenze specializzate di comprendere i risultati.

2.1.2 SMAT

Il Sistema di Monitoraggio Avanzato del Territorio (SMAT) è un progetto per lo studio di un sistema di monitoraggio avanzato del territorio Piemontese con l'obiettivo di prevenzione e controllo di diversi scenari, sia naturali che non, mediante l'utilizzo di UAV.

Gli scenari di utilizzo possono essere ad esempio situazioni di alluvione, incendio frane ma anche traffico, inquinamento, coltivazioni o attività illegali sul territorio.

Il sistema è composto da tre elementi fondamentali (Figura 2.5):

- UAS, flotta di UAV di diverse tipologie che operano a diverse altitudini, prestazioni e capacità
- SC&SSC (Segmento di Comunicazione & Stazione di Supervisione e Coordinamento), elemento chiave nella fornitura dei dati richiesti dagli utenti e nella opportuna traduzione delle loro esigenze in termini operativi.
- Networking, ovvero la comunicazione tra i vari soggetti e l'integrazione dei UAV nello spazio aereo

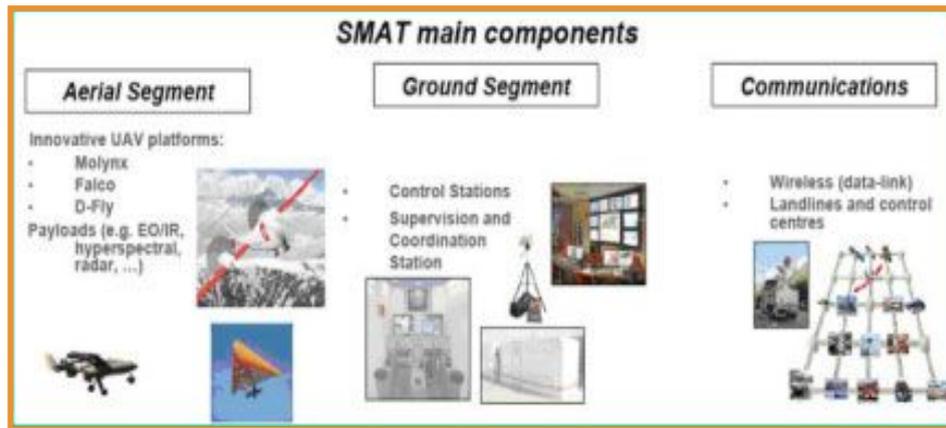


Figura 2.5: componenti principali del progetto SMAT

Nello specifico l'azienda ALTEC aveva il compito di progettare e definire la SSC, nodo centrale dell'infrastruttura SMAT che consente agli operatori di ricevere ed elaborare i dati inviati dagli UAV per ogni missione od obiettivo e, a loro volta, inviare specifiche indicazioni alle specifiche piattaforme UAV. Questo ha comportato tra le varie mansioni anche lo sviluppo di un'interfaccia (Figura 2.6) per la progettazione dei voli ed un sistema di mission control per l'acquisizione dei dati degli UAV in tempo reale ed una prima analisi visiva.

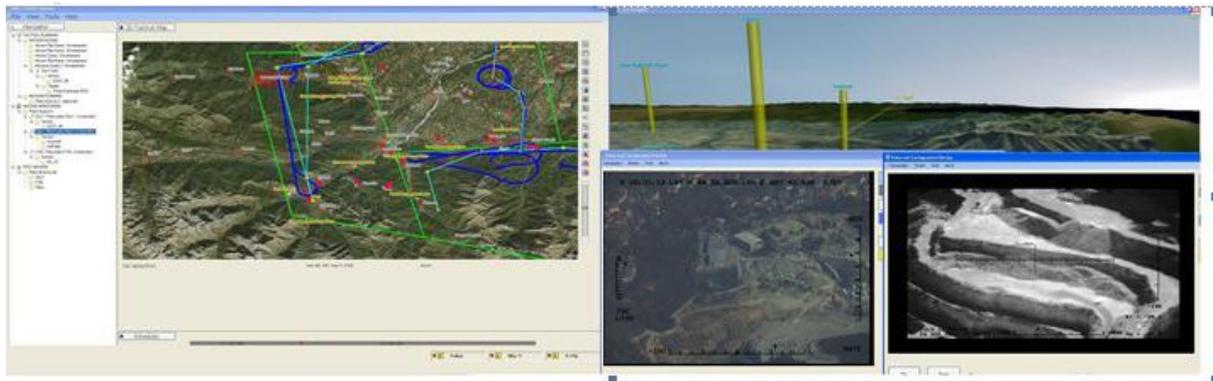


Figura 2.6: interfaccia per progettazione voli

Nell'ambito di questa area di progetto, con il presente lavoro si vuole esplorare la possibilità di utilizzare i nuovi strumenti di visualizzazione olografica e Mixed Reality a supporto dell'attività decisionale e operativa degli Utenti coinvolti in operazioni sul territorio.

2.1.3 Virtual, Augmented e Mixed Reality

La fusione tra mondo digitale e mondo reale con la conseguente immersione dell'utente finale in una "nuova realtà" con pieno accesso ad entrambe senza più necessità di strumenti intermediari sta subendo una forte accelerazione negli ultimi anni sia in ambito privato che in quello industriale.

E' utile, a questo proposito distinguere le tre tipologie di fusione tra le due realtà, quella digitale e quella fisica, definite fino ad oggi:

1) **Virtual Reality**

La VR si propone di immergere il fruitore in un ambiente totalmente digitale perdendo la percezione dell'ambiente circostante. A tale scopo vengono utilizzati dei visori che occludono totalmente la visione in abbinata a degli apparati acustici così che il soggetto riceva esclusivamente input sia visivi che uditivi generati dal computer e possa interagire con il questo ambiente tramite controller tattili.

2) **Augmented Reality**

La AR aggiunge elementi digitali al mondo reale. Mantenendo al centro dell'attenzione quest'ultimo, lo arricchisce di informazioni digitali aggiungendo uno strato percettivo.

3) **Mixed Reality**

La MR mette insieme i due modi, digitale e reale, rendendo possibile la manipolazione di entrambi. Mettendo assieme le tecnologie di sensoristica e quella dell' image processing permette di immergersi e vedere la realtà fisica intorno manipolando simultaneamente quella digitale senza l'uso di controller aggiuntivi, se non il propri movimenti, abbattendo la barriera tra le due.

Anche se si tende a considerarle tecnologie nuove, sviluppate solo negli ultimi anni, in realtà trovano la loro origine verso la fine degli anni '50. Più precisamente nel 1957 Morton Helig costruì una macchina chiamata Sensorama, con l'intento di creare un'esperienza cinematografica immersiva per il fruitore. Infatti il macchinario proiettava un ambientazione in una specie di 3D stereoscopico. Anche se non ebbe successo fu comunque il primo tentativo di realtà virtuale.

Nel 1966 Ivan Sutherland^[6], professore di ingegneria elettronica all'università di Harvard, costruì il primo modello di uno dei più importanti device sia per la realtà virtuale che per la realtà aumentata, tutt'oggi ancora usato. Il device era il Head Mounted Display o HMD ovvero una specie di casco che occludeva la vista dell'utilizzatore mediante degli schermi. Il primo modello era talmente grande che non poteva essere sorretto da una persona, veniva invece sospeso dal soffitto del laboratorio. Allo stesso tempo la potenza di calcolo erano inversamente proporzionali al peso, le capacità grafiche si limitavano ad una visualizzazione wireframe della ambientazione generata. Ad ogni modo fu il primo passo verso i caschi/occhiali per la AR e VR presenti oggi.

Come si diceva il concetto di AR circolava già nell'ambito tecnologico ma la nascita del termine Augmented Reality^[7] si attribuisce al professore Tom Caudell mentre lavorava al Computer Services' Adaptive Neural Systems Research and Development project di Seattle. Durante la sua ricerca per semplificare i processi di produzione ed ingegnerizzazione per la compagnia aerea mediante l'uso della realtà virtuale sviluppò un software che riusciva a visualizzare un'immagine sintetizzata ad una reale, mostrando così nel processo di assemblamento dove andavano messi determinati cavi. Così facendo i meccanici che lavoravano in produzione non dovevano fermarsi per consultare gli schemi cartacei o chiedere a personale più informato, riducendo i tempi e le possibilità di fraintendimento.

Nello stesso periodo, intorno al 1992, altri due gruppi di ricercatori fecero enormi passi in avanti. LB Rosenberg creò il primo vero e proprio sistema di AR per l'Air Force degli Stati Uniti chiamato VIRTUAL FIXTURE.

Il secondo gruppo composto da Steven Feiner, Blair MacIntyre e Doree Seligmann, i quali ora ricoprono tutti posizioni di spicco nell'ambito dell'AR, avevano prototipato un sistema basato su AR chiamato KARMA adibito a funzioni di assistenza e manutenzione. Nella fattispecie il team aveva costruito un HMD, con la collaborazione di Logitech, per il supporto alle stampanti. Il sistema doveva mostrare delle immagini 3D "fantasma" (non ancora propriamente degli ologrammi) che mostrassero come e dove mettere i fogli nella stampante senza l'utilizzo di manuali o istruzioni esterne.

Il vero cambio di marcia nella diffusione ed utilizzo di queste tecnologie si ha a cavallo tra il 1999 e il 2000 quando Hirokazu Kato del Nara Institute of Science and Technology rilascia ARToolKit con licenza open-source^[8]. Per la prima volta, permise di combinare e allineare insieme un video ripreso da una telecamera con elementi grafici 3D creando inoltre un'infrastruttura che poteva appoggiarsi su qualunque sistema operativo.

Nel 2008 T-Mobile rilascia la prima applicazione per android che portò la realtà aumentata sui cellulari che mostravano degli oggetti 3D quando si puntava la camera su determinati punti sensibili.

Negli anni a seguire si svilupparono sempre più avanzate applicazioni in AR e, contemporaneamente, nasce una nuova generazione di HMD per la VR, accessibili al pubblico, pensati soprattutto per il gaming.

Agli inizi del 2015 avviene un'altra svolta. Dai laboratori di Redmond appare per la prima volta ufficialmente il progetto HoloLens di Microsoft (Figura 2.7). Vi erano ovviamente altri competitor che stavano sviluppando tecnologie simili nel mentre (es. MagicLeap), ma Microsoft fu la prima a rilasciare un device funzionante per la MR.



Figura 2.7: HoloLens

L'HoloLens è un HMD, see-through, stand-alone. Ovvero è un visore che non necessita nessun dispositivo aggiuntivo per funzionare, differenza significativa rispetto ai HMD precedenti sviluppati. Permette di vedere l'ambiente circostante, a differenza dei visori per VR, sovrapponendoci ologrammi 3D capaci di interagire con la realtà fisica distinguendosi dalla AR e coniando il concetto di mixed reality. Il progetto nasce dagli stessi sviluppatori del Kinect spingendo la tecnologia del riconoscimento dell'ambiente circostante e dei comandi tramite gesti a nuovi orizzonti. Il sistema infatti non si appoggia a nessun controller esterno ma basa i suoi metodi di input sul riconoscimento di una serie di gesti ben codificati e sul riconoscimento vocale. Oltre alla più ovvia apertura a nuove filosofie di applicazioni capaci di riconoscere ed interagire con il mondo circostante, segna un cambiamento sostanziale anche nella concezione di User Interface. Tutti i moduli che compongono le classiche UI non hanno più la limitazione di uno schermo ed acquisiscono proprietà tridimensionali e con esse la possibilità di essere pienamente configurabili nello spazio e poter reagire attivamente a

quest'ultimo. Anche il sistema di puntamento si sgancia da un device esterno, viene invece gestito da gaze, o sguardo, dell'utente.

2.1.4 Intel “commercial drone solutions for enterprise”

Intel ha creato negli ultimi anni un sistema che include l'intera catena di raccolta, processing e fruizione dei dati, al fine di fornire alle aziende un servizio ottimizzato dall'inizio alla fine per la ricognizione mediante droni. Il sistema è composto da tre componenti: i droni, Mission Control Software, Insight Platform^[9].

I droni:

L'hardware dei droni è progettato e prodotto da intel per essere “seamlessly integrated” con il software e la piattaforma proprietari. Intel offre due tipologie di UAV. Un multi-elica per la raccolta dati con accuratezza millimetrica. Ed uno a fixed-wing con lunga autonomia e possibilità di voli completamente autonomi per coprire grosse aree o raggiungere luoghi difficilmente accessibili.

Intel Mission Control:

Il software permette di creare piani di volo sia in 2D che in 3D per agrimensura, mapping ed ispezioni, anche su oggetti tridimensionali come edifici e torri. Potendo anche attingere ad una serie di pattern di volo prefatti. Una volta creato è possibile fare un controllo che il piano di volo rispetti tutti i parametri di sicurezza e simulare la risposta dei sistemi di bordo, come camere e sensori.(Figura 2.8)

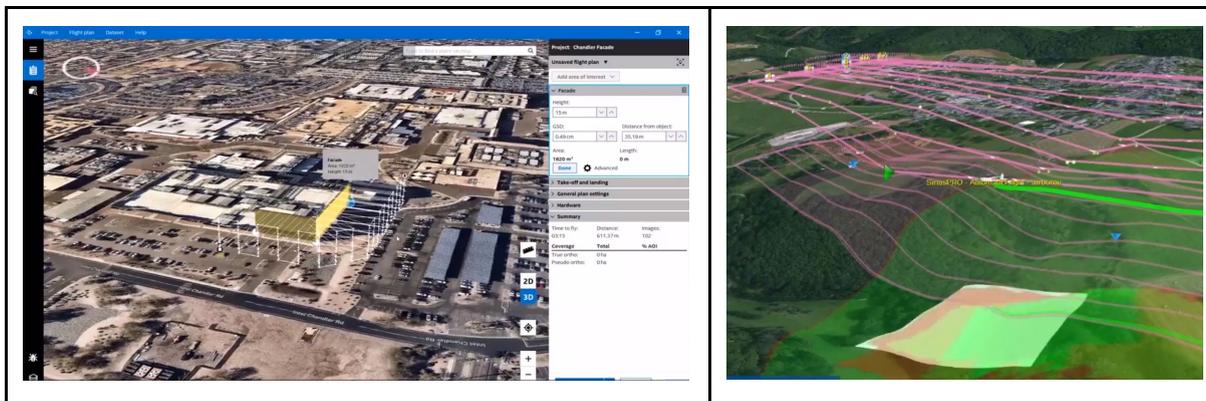


Figura 2.8: Screenshot del Mission Control di Intel

Intel Insight Platform:

La piattaforma, basata su cloud, fornisce la gestione dei dati asset-based ed un engine per il processamento delle fotografie di rilievo che permette di creare e gestire i digital twin dei propri asset. Mette inoltre a disposizione una serie di strumenti per l'analisi ed il monitoraggio che trasformano le immagini, i modelli 2D e quelli 3D in “insights “ ovvero in oggetti “approfonditi”, che evolvono nel tempo, facilitandone la manutenzione. Basandosi su cloud, la piattaforma facilita la condivisione di dati, commenti e report con partner o clienti. (Figura 2.9)

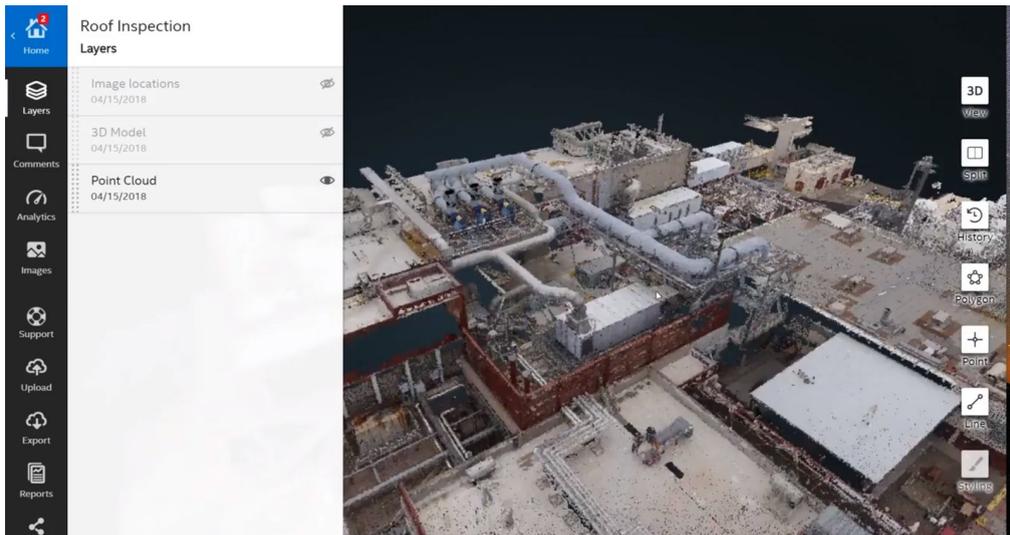


Figura 2.9: Screenshot della piattaforma Insight di Intel

2.2 Valutazioni economiche

Nel corso dell'ultimo anno (2017) il mercato del AR/VR ha fatto registrare il suo record negli investimenti raggiungendo i 3 Miliardi di dollari, di cui circa la metà solo nell'ultimo quadrimestre. Viene dunque calcolata una crescita del 60% nel corso dei prossimi anni. Si può inoltre notare come delle 28 macrocategorie di mercato quelle con maggior investimenti sono quella "Medical" e "Tech" (Figura 2.10).

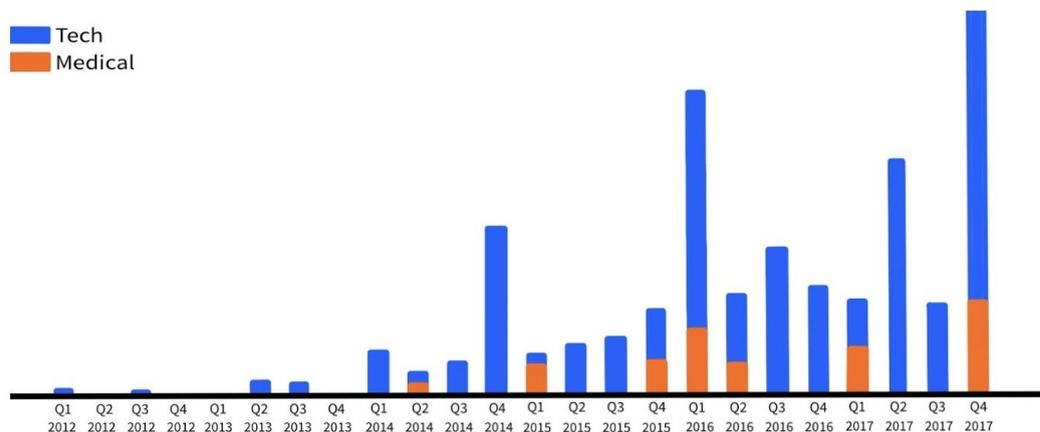


Figura 2.10 Investimenti nel AR/VR

Nello specifico dell'Italia nel 2017 l'industria 4.0, secondo i dati diffusi dall'osservatorio del Politecnico di Milano, ha raggiunto un valore di 2,4 miliardi. Una crescita rispetto al 2016 del 30%^[10]

Per quanto riguarda i droni, gli investimenti negli ultimi anni sono cresciuti altrettanto in fretta. In questo caso sono più le startup ad essere cresciute o acquistate. Nel 2015 si

registra una crescita di capitali delle startup di 172 milioni di dollari, con un incremento del 61% rispetto all'anno precedente solo nei primi due quadrimestri^[11]

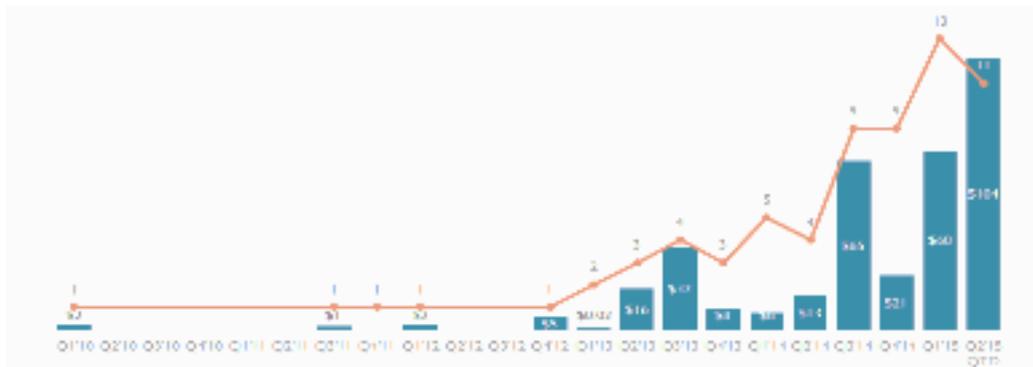


Figura 2.11 Investimenti nei droni

Il mercato dei droni non comprende solamente il discorso manufacturing. L'incremento dell'utilizzo di droni come strumenti di monitoraggio e raccolta dati porta con se lo sviluppo di aziende per la gestione ed analisi di questi dati. Espandendo così il mercato in: Terrestrial Imagery & Mapping, Precision Agriculture, Inspection & Monitoring, Delivery & Transport, Entertainment, Airspace Management, Military & Defense, Navigation & Autonomy. L'insieme di queste aziende nel primo quadrimestre del 2017 ha fatto registrare un totale di investimenti di 113 milioni di dollari. La prima compagnia in termini di finanziamenti che non sia nell'ambito manufacturing, si occupa di Terrestrial Imagery & Mapping e ha fatto registrare una crescita di quasi 61 milioni di dollari partendo da \$40.16M del 2015 a \$101M nel 2017 con un incremento del 152%

3 Requisiti del progetto

3.1 Obiettivi progetto

Lo scopo del progetto di tesi è quello di valutare le possibilità applicative della Mixed Reality nell'ambito del controllo territoriale e sviluppare una prima applicazione per HoloLens.

A valle di una fase iniziale di ipotesi alternative, l'ambito applicativo scelto come spunto per questa esplorazione è la "Situation Room"¹ a supporto della valutazione di scenari "analitici" o "critici" territoriali. In questo contesto applicativo vengono gestite le informazioni rilevanti di un dato Scenario Operativo territoriale facilitando all'utente la vista sinottica delle informazioni rilevanti, la loro georeferenziazione e il successivo arricchimento con informazioni testuali/grafiche e/o olografiche.

Nella versione finale, l'applicazione dovrà fornire all'utente la possibilità di avere un'immagine olografica tridimensionale dello scenario operativo, con l'integrazione di informazioni provenienti da diverse sorgenti, quali in particolare: rilevamenti provenienti da UAV, o altri mezzi "unmanned", operanti nello scenario, posizionamenti di unità operative provenienti da altri sistemi di supervisione e controllo o da segnalazioni individuali di posizionamento, marcaggi e segnaletiche originate da altri operatori, che collaborano, mediante l'utilizzo di altri visori, sullo stesso scenario operativo e la possibilità di configurare liberamente il proprio "spazio di lavoro" mediante la mixed reality.

Dopo una prima analisi degli strumenti tecnologici a disposizione e una valutazione di effort e tempi necessari a perseguire i desiderata complessivi, si è resa evidente la necessità di optare per un approccio progressivo verso l'obiettivo e a perseguire la realizzazione con un tecnica incrementale, questo sia per gestire l'aspetto quantitativo delle implementazioni necessarie, sia per la necessità di valutare le scelte funzionali intraprese in corso d'opera. Utilizzando approccio di "Iterative Development" è stato individuando per la prima iterazione, oggetto di questo lavoro, un nucleo di prestazioni che potesse costituire da un lato una insieme compiuto di funzionalità e dall'altro la base tecnologica/ infrastrutturale per le successive evoluzioni funzionali.

Le funzionalità principali della prima iterazione sono state individuate in:

- Generazione olografica del territorio operativo;
- Possibilità di elaborare dati provenienti da più sorgenti (locali ed esterne) e con diversi sistemi di coordinate (WGS84, UTM);
- Integrazione di dati di volo degli UAV;
- Possibilità di creare e valutare i parametri di un nuovo volo/missione;
- Possibilità di aggiungere alle informazioni tridimensionali del terreno informazioni di altra natura utili all'analisi situazionale;

¹ "Situation Room" intesa come luogo, eventualmente virtuale, di concentrazione/convergenza di informazioni relative alla gestione di uno scenario a supporto dei successivi processi decisionali e di coordinamento operativo.

- Elementi di UI e UX in Mixed Reality.

Durante lo sviluppo si è deciso di chiamare l'applicazione "Holographic Situation Room" o, in breve, HSR.

4 Progettazione e Sviluppo

4.1 Vista di contesto

Il diagramma in figura (Figura 4.1) illustra il contesto applicativo del sistema a regime in cui si è immaginato l'utilizzo del HSR.

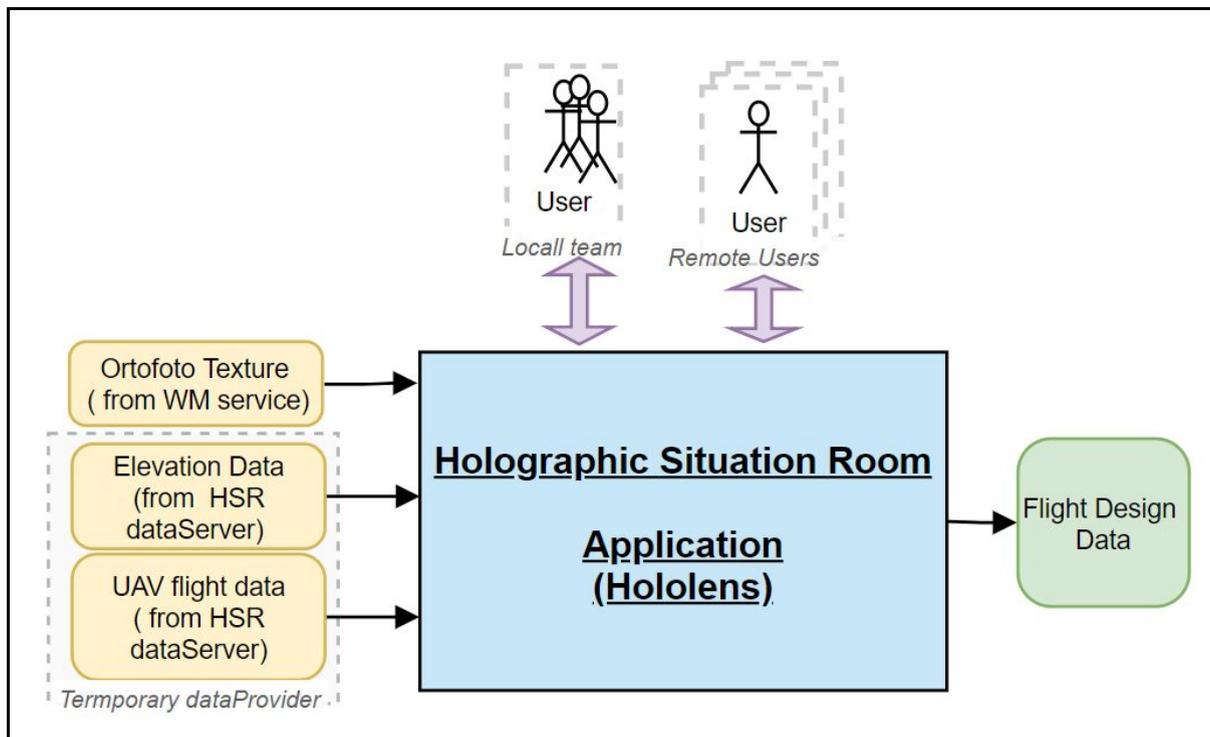


Figura 4.1 vista di contesto

Con riferimento al diagramma si evidenziano i seguenti elementi:

4.1.1 Utilizzatori della applicazione

Il Sistema HSR prevede l'utilizzo di una istanza individuale della applicazione da parte di un utente sul proprio visore. Si ipotizza, in futuro, anche l'utilizzo da parte di più utenti ognuno con il proprio visore. In questo caso si prevedono due scenari. Un utilizzo da parte di più utenti co-locali che condividono la visione del medesimo ologramma, oppure utilizzatori non co-locali (e.g: operatori sul campo), che condividerebbero indipendentemente le informazioni del medesimo scenario operativo.

4.1.2 Dati in ingresso

Il sistema HSR prevede l'acquisizione di dati elementari da molteplici fonti. I dati acquisiti e utilizzati dal sistema sono diversi per ruolo che per formato e metodo di accesso.

Nel seguito i dati sono illustrati in funzione del ruolo svolto nella applicazione, e successivamente per "tipologia tecnologica" (i.e: metodo di acquisizione, tipo di codifica, etc...)

- Dati altimetrici dei terreni: sono importati nel server locale come GeoTIFF con sistema di coordinate WGS84 o UTM 32 Nord. All'interno del server vengono codificati mediante le librerie GDAL in file raster ASCII con metadati trasformati nel sistema di coordinate WGS84 nell'eventualità che non lo siano già.
- Ortofoto dei terreni: dipendentemente dalla tile del terreno interessata viene creata una query per il servizio WMS del GeoPortale Piemonte. Una volta interrogato il WMS, in base ai parametri impostati, risponde con un'immagine PNG 512x512 e 96 DPI.
- Dati dei voli: i voli vengono importati nel server locale come file GPX

4.1.3 Dati Geografici di contesto:

Un primo insieme di dati gestiti è costituito dai dati geografici del contesto operativo che funge da "base" per tutte le altre informazioni gestite².

Sono previsti due fonti di dati:

- i dati di conformazione altimetrica del territorio, che forniscono le informazioni di "forma" della superficie del terreno;
- i dati fotografici territoriali, che facilitano la *interpretazione* e la *contestualizzazione* dei dati di conformazione altimetrica;

In generale le fonti di questi due dati sono indipendenti, sia come origine che come riferimento temporale³, ed è l'applicazione a provvedere alla correlazione delle due insiemi di dati tramite i metadati di georeferenziazione.

Relativamente ai dati di conformazione altimetrica sono stati gestiti sia dati "pubblici" di riferimento per la Regione Piemonte⁴ sia dati proprietari di rilevamenti "sperimentali"/"mirati" da parte di UAV. In entrambi i casi il repository si trova su un server locale.

I dati forniti dalla Regione Piemonte sono l'archivio completo di file GeoTIFF che compongono il DTM del Piemonte derivanti da delle riprese aeree fatte dalla regione tra il 2009 e il 2011⁵. Il DTM copre tutto il territorio regionale ed è stato acquisito con metodologia uniforme (LIDAR) in standard di livello 4. La risoluzione della griglia è di 5 m, con una precisione in quota di $\pm 0.30\text{m}$ che può arrivare ad un $\pm 0.60\text{m}$ nelle aree di minor precisione,

² Nel seguito riferito questo insieme di dati verrà riferito anche come "il Terreno"

³ Potenzialmente i due insiemi di dati possono avere delle discrepanze dovute a eventi intercorsi tra i due rilevamenti, ma nella pratica questo aspetto non costituisce di norma un problema dal punto di vista pratico in quanto ben compensato dalla capacità interpretativa dell'Utente. (e.g: riempimento/svuotamento di bacini, eventi di frana, livellamenti artificiale/manufatti, etc. ...)

⁴ /* riferimento ai due insiemi: DTM5 nuovo e quello "vecchio", con "nome-e-cognome" del rilevamento. */

⁵ ultima e più recente ricostruzione del Piemonte come dati pubblici

corrispondenti alle aree boschive e densamente urbanizzate. La tecnologia LIDAR (Figura 4.2), acronimo di Light Detection and Ranging o Laser Imaging Detection and Ranging^[12], è una tecnica di telerilevamento “attivo” per l’esecuzione di rilievi topografici ad alta risoluzione. Si basa su una tecnica di scansione laser che restituisce una nuvola di punti, o point cloud, che rappresentano le coordinate tridimensionali del punto scannerizzato. Questa point cloud viene georeferenziata mediante l’interpolazione con dati GPS presente sul veivolo.



Figura 4.2 rappresentazione tecnologia LIDAR

Peri i dati fotografici si utilizza il servizio WMS (Web Map Service) della Regione Piemonte⁶. Nella fattispecie queste ortofoto fanno parte dello stesso progetto di rilevazioni sul territorio del 2009-2011. Sono state scattate durante i rilevamenti Lidar per garantire la coerenza tra i due dataset.

Tramite il suddetto servizio WMS è possibile effettuare la richiesta di un’immagine mediante una query HTTP. Nella richiesta è possibile specificare l’area di interesse, il livello di zoom e il formato dell’immagine di risposta. Le aree di interesse devono essere espresse in WGS84 a differenza dei DTM che vengono forniti in UTM.

I dati di UAV provengono da voli di UAV di Alenia.

- **Tracciati di volo degli UAV:**

Questo insieme di dati è costituito da data-set che forniscono una sequenza di “posizioni” di un UAV, rilevate durante il volo. I dati di volo sono dati “statici”, acquisiti da HSR “a posteriori” rispetto al volo⁷ e, nel contesto della applicazione, sono mirati a consentire la

⁶ http://geomap.reteunitaria.piemonte.it/ws/taims/rp-01/taimsortoregp/wms_ortoregp2010

⁷ Allo stato attuale i dati progressivi di posizionamento dell’ UAV non sono resi disponibili in real time dal sistema di controllo del volo, ma sono disponibili i dati complessivi del tracciato

individuazione del sottoinsieme di aree coperte dai rilevamenti effettuati dagli UAV⁸. In generale questi dati sono forniti dai sistemi di monitoraggio del volo degli UAV a volo concluso

- Dati di geoposizionamento di entità rilevanti nel contesto dello scenario:

Questo insieme raccoglie due principali sottogruppi: “dati statici” (e.g: eventuali infrastrutture notevoli/sensibili nel contesto dello scenario) e “dati dinamici” (e.g: posizionamento di unità mobili, (squadre, mezzi, focolai, etc, ...) nel contesto dello scenario)

- **Dati Statici:** Sebbene i “Dati Statici” possano in linea di principio essere originati/acquisiti da cartografia tematica, in HSR sono gestiti come marcaggi segnalati manualmente dall’Utente in modo da consentire la selezione e segnalazione delle sole informazioni ritenute rilevanti per lo scenario e contesto operativo in oggetto, e mantenere lo scenario focalizzato su un ridotto insieme di dati ben visibile/distinguibile, e prevenire il “cluttering” di informazioni. Un particolare sottocaso di questo tipo di dato è costituito da “paline di segnalazione” testuali con cui un Utente può notificare/comunicare messaggi di informativa/allerta contestualizzati ad uno specifico luogo dello scenario operativo.
- **Dati Dinamici:** I dati “Dati Dinamici” sono resi disponibili al sistema HSR da un insieme di fonti possibilmente eterogeneo. Possibili fonti dati sono sistemi di Fleet-Management o sistemi di Supervisione e Controllo di altri centri Operativi. Allo scopo di facilitare l’integrazione con questo insieme di sorgenti potenzialmente eterogeneo, e considerate le caratteristiche di moderata “dinamicità” del dato, si è mantenuta una modalità di integrazione “loosly-copuled” per questi dati. In particolare è previsto che le sorgenti dati possano fornire e aggiornare dati di posizionamento georeferenziato di singoli oggetti caratterizzati da una quaterna “Tipo”, “Identificativo”, “posizione-georeferenziata”; “timeStamp” tramite un apposito WEB Service esposto da HSR. I dati così segnalati sono visualizzati sull’HSR Client con apposite icone, nell’ambito di un refresh periodico dello scenario visualizzato. E’ altresì previsto su HSR un meccanismo di acquisizione attiva di analoghi data-element, tramite polling periodico ad una lista di Web Services configurabile sul sistema.

di volo prodotti, con un apposito post-processing, dal sistema di controllo dell’UAV, a volo ultimato.

⁸ La localizzazione/tracciamento in tempo reale degli UAV esula dagli scopi prefissati per la applicazione HSR ed è gestita dagli appositi sistemi di monitoraggio di ciascuna tipologia di UAV.

4.2 Vista funzionale

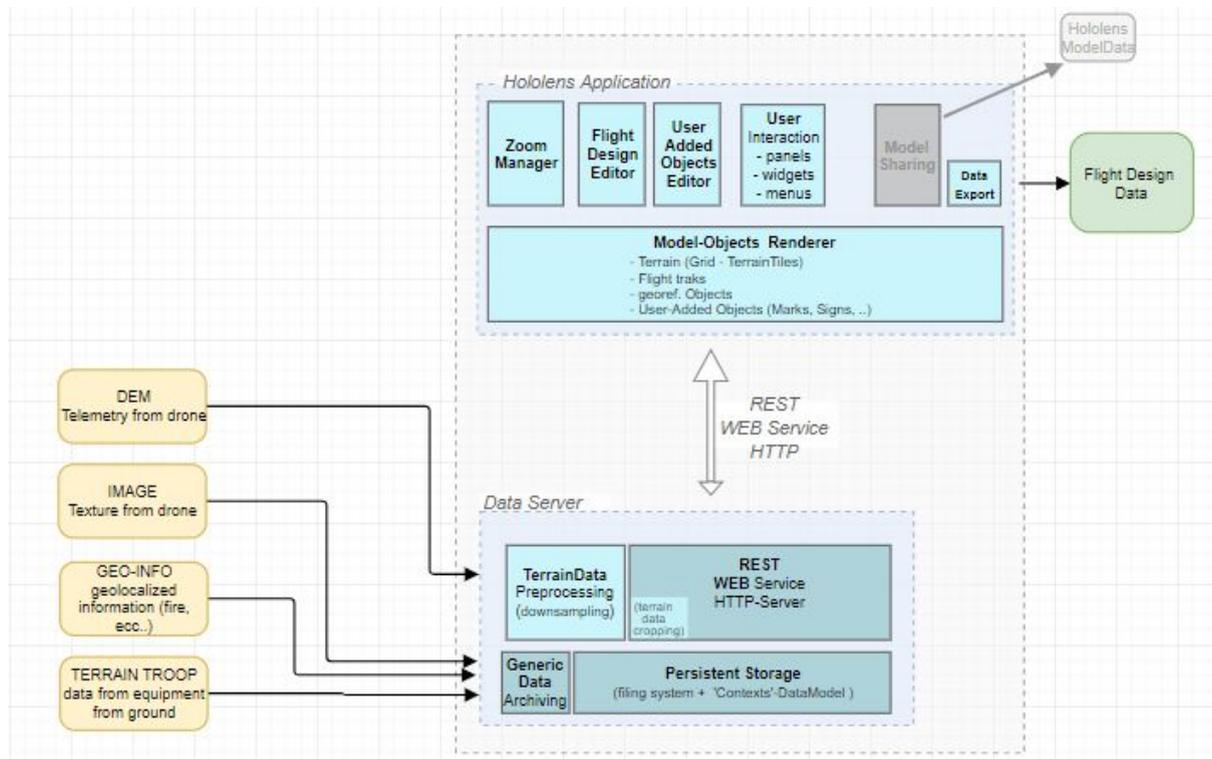


Figura 4.3 vista funzionale

4.2.1 Visualizzazione dinamica dei terreni

L'applicazione permette di visualizzare e caricare dinamicamente qualunque DTM del Piemonte. Attualmente si usa una collezione di DTM con risoluzione 5 metri realizzata tra il 2009 e il 2011. Il DTM viene passato all'applicazione come conversione ASCII di GeoTIFF e visualizzato mediante una griglia virtuale 3x3 tile. Le tile possono avere una risoluzione arbitraria mostrando porzioni maggiori o minori del terreno. In questo caso con risoluzione si intende contenere più o meno valori della matrice derivante dal GeoTIFF, codificati a loro volta dai pixel.

Visualizzando una porzione finita del DTM totale è possibile navigare quest'ultimo mostrando quadranti diversi (Figura 4.4). Questo avviene mediante i bottoni di navigazione descritti sopra. In questo caso non si voleva ricorrere al caricamento di un'altra scena rallentandone la fruizione. Allo stesso tempo si voleva avere la possibilità di usare DTM ad alta risoluzione comportando un incremento di vertici nella visualizzazione del medesimo settore. Si è dunque sviluppato un metodo di caricamento e distruzione dinamico delle tile nella griglia. In questo modo il numero dei vertici elaborati dall'HoloLens rimane costante ma ci si può muovere dinamicamente spostando la griglia 3x3.

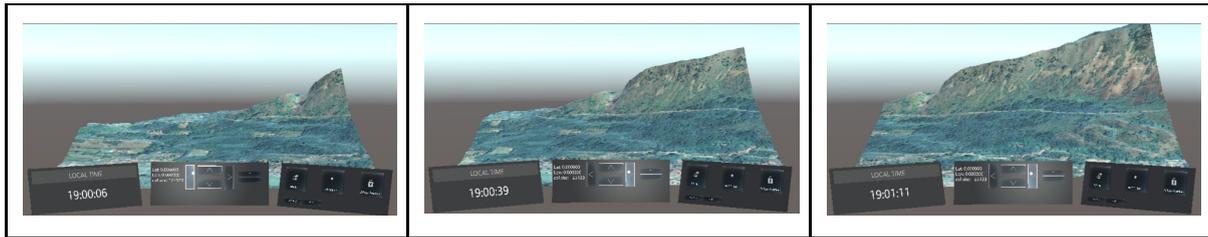


Figura 4.4 navigazione terreno

4.2.2 Visualizzazione Layer tematici

Questa funzione permette di visualizzare informazioni diverse sulla mesh del terreno. Le tipologie di informazioni si dividono in due categorie.

La prima applica immagini o texture, derivanti da terze parti. Nella fattispecie permette di applicare sul terreno le ortofoto del geo portale del piemonte. Ancorché non implementato, con lo stesso metodo, è possibile applicare una qualunque immagine fornita dal WMS.

Un sottoinsieme di questa categoria è la sovrapposizione di immagini anche parziali. Ovvero la possibilità di aggiungere al terreno con le ortofoto un'immagine, ad esempio, di una termocamera. Quest'ultima non sarà grande come tutta la mesh ma potrebbe interessarne solo una parte di essa. Per poter sovrapporre le due immagini continuando a garantire la georeferenziazione di queste si è dovuto sviluppare uno shader che gestisse ed interpolasse due immagini e due uv map su una stessa mesh.

Con questa si apre il secondo insieme di informazioni visualizzabili sul terreno, ovvero quelle informazioni calcolate dinamicamente mediante gli shader. In questa categoria rientrano la mappa delle altimetrie e la mappa delle pendenze. Queste infatti non si rifanno a dati esterni ma calcolano i valori dinamicamente basandosi sul modello digitale del terreno.

La mappa delle altezze converte in codice colore l'altezza di ogni singolo vertice della mesh. Non ci si basa su dati esterni per poter personalizzare la scala delle altezze consentendo all'utente di creare una scala colore relativa alla porzione di terreno visualizzato o anche solo di una porzione d'interesse. I valori di massimo e minimo rimangono quelli impostati anche mentre si naviga nel terreno permettendo di confrontare aree non visualizzabili simultaneamente.

La mappa delle pendenze, a sua volta, permette di convertire in codice colore l'inclinazione rispetto al piano orizzontale di ogni vertice. Dato utile più per eventuali operazioni da terra che per i voli. Può però aiutare a valutare la necessità di più passaggi o angoli diversi per la corretta cattura dei dati da parte dei sensori dell'UAV (Figura 4.5).

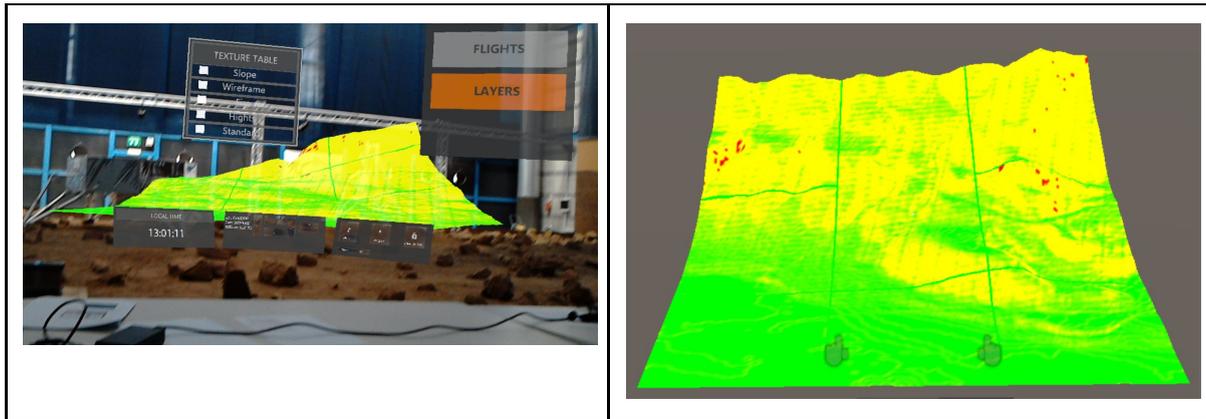


Figura 4.5 Slope map prima su Hololens e poi su simulatore

Anche se ad oggi non implementate, la struttura di queste funzionalità è pensata per acquisire e gestire i dati in maniera dinamica. Le due funzionalità in tempo reale che si vorrebbero raggiungere in un futuro e che hanno condizionato alcune scelte implementative sono:

- La visualizzazione di output di termocamere georeferenziate in tempo reale sovrapposte al modello del terreno e alle ortofoto;
- La possibilità di visualizzare e confrontare diverse scansioni e le mesh risultanti di una medesima zona. Permettendo di valutare modifiche del territorio quali cambi di pendenza o altezza. Tutto questo in tempo reale.

4.2.3 Visualizzazione dei voli UAV

Mediante un elenco è possibile visualizzare i voli effettuati precedentemente. Per ogni volo è possibile visualizzare il percorso, i vari way point, le proiezioni sul terreno degli scatti effettuati e simulare nuovamente il volo.

Dovendo valutare un passaggio in una zona già coperta in passato si può in questo modo avere un riscontro visivo di tutte le informazioni principali già acquisite in precedenza. Durante la progettazione del nuovo volo è così possibile valutare migliorie o evitare passaggi inutili in maniera visiva ed intuitiva senza dover andare ad incrociare dati da varie fonti.

Ovviamente questi voli non presentano nessuna impostazioni se non le opzioni di visualizzazione o meno delle varie informazioni.

Anche nello sviluppo di questa sono stati tenuti a mente eventuali sviluppi futuri. I way point vengono caricati in maniera dinamica. Nella simulazione del volo l'UAV calcola il suo percorso dinamicamente. Al raggiungimento di un way point il software controlla se ne esiste uno successivo per procedere con la simulazione o fermarsi. Allo stato attuale si potrebbe calcolare direttamente il percorso per intero ma caricando i way point in maniera progressiva si lascia aperta la possibilità di avere un flusso dati in real time e poter visualizzare il volo di un UAV mentre avviene aggiungendo così la funzionalità di mission control

4.2.4 Impostazione di una ricognizione UAV

Valutando i punti critici di una ricognizione mediante UAV sono stati riscontrati due macro argomenti, Traiettorie e Field Of View del sensore. Il primo vuole valutare il passaggio fisico del veivolo nello spazio aereo tenendo conto di altitudini, velocità e spazio percorso. Il secondo serve a valutare le esigenze del sensore per l'acquisizione corretta di dati come ottica, sovrapposizione immagini, numero di passate, ecc.. Allorchè dipendenti l'uno dall'altro con la necessità di essere gestiti con due interfacce diverse.

Mediante questa funzione si vuol dare la possibilità all'utente di valutare tutti i parametri di una nuova ricognizione (Figura 4.6)

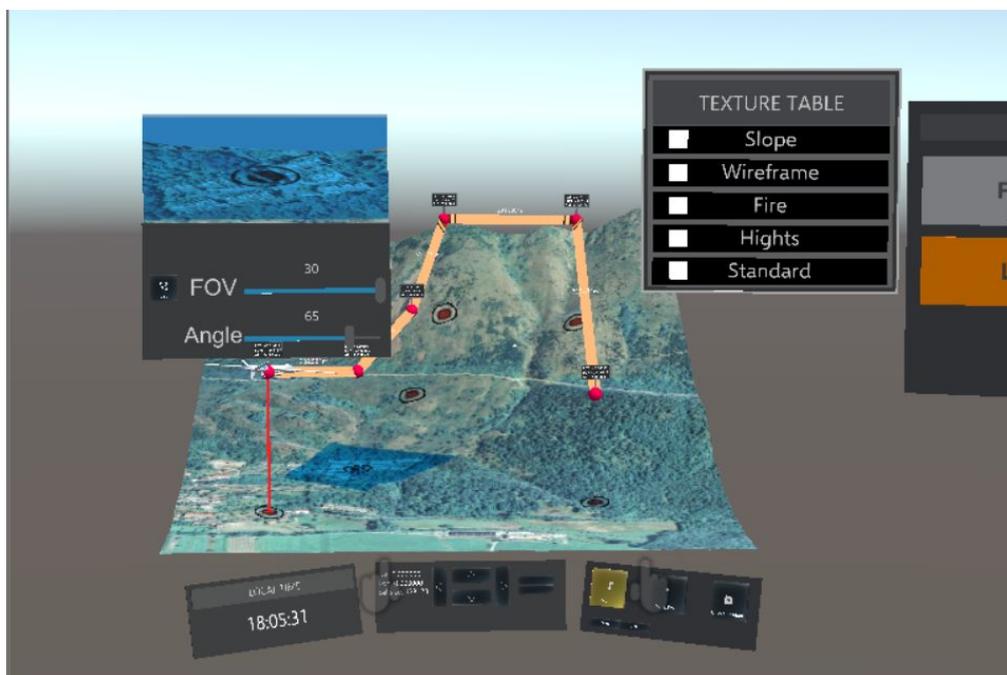


Figura 4.6 esempio progettazione volo

4.2.4.1 Traiettorie

Mediante questa funzionalità è possibile progettare e valutare un nuovo volo sul territorio. Le prime azioni da vedere sono la creazione dei way point. A questo riguardo, una volta selezionato "New WP" è possibile selezionare un punto sul terreno per fissare le coordinate del way point. Si vedrà comparire un oggetto che simboleggia il wp con una tooltip che visualizza le coordinate del punto. Oltre al wp viene creato un collider rettangolare non visibile sul piano virtuale perpendicolare al terreno centrato sul punto. In questo modo è possibile impostare l'altezza del wp, l'utente potrà muoversi liberamente senza dover prestare attenzione a rimanere sulla verticale del punto poichè verrà letta solo il valore della y del cursore.

Mentre si imposta l'altezza, per aiutare visivamente l'utente, viene proiettata una linea tra il wp ed il terreno. Impostata anche l'altezza la linea sparisce e il collider viene distrutto, dando di nuovo piena libertà di movimento al cursore. Rimane comunque un simbolo sul terreno, paragonabile ad un'ombra, della proiezione del wp su di esso in modo da lasciare un riferimento per l'operatore durante la selezione del wp successivo. Dalla selezione del secondo wp in poi viene costruito anche il tragitto dell'UAV rappresentato come una linea

che unisce i vari way point. Su ogni tragitto viene visualizzata, in maniera dinamica, la distanza tra i due punti. (Figura 4.7)

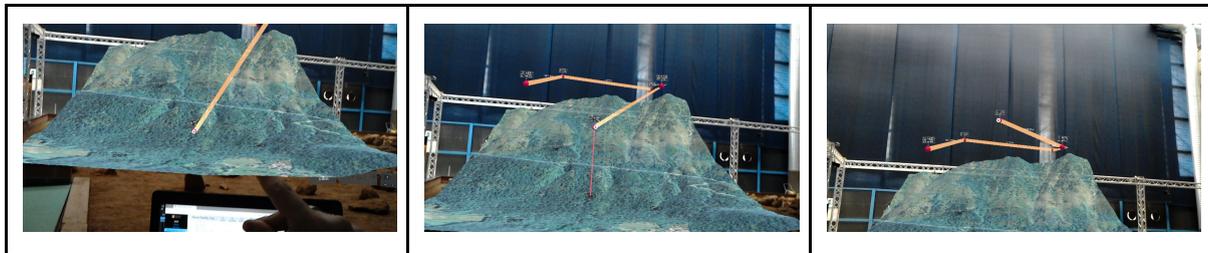


Figura 4.7 Sequenza di creazione di un way point

Oltre a creare nuovi way point da zero, è possibile crearne copiando l'altimetria di quello precedente accorciando di processo di creazione. Basta infatti selezionare la modalità "Copy WP", che automaticamente deselezionerà il "New WP". Selezionando ora un punto su terreno verrà direttamente creato un nuovo wp con altezza pari a quella precedente. Se si volesse dunque prevedere un volo a quota costante l'operazione si velocizzerebbe senza che l'utente presti attenzione a impostare la medesima altezza.

Sui way point creati è possibile fare tre azioni: modificare l'altezza, modificare le coordinate o eliminarli. Modificare l'altezza viene gestita come la prima impostazione di questa. La modifica delle coordinate usa lo stesso principio ma questa volta il collider è parallelo al terreno e permette all'utente di muovere il wp solo sul piano orizzontale. In fine la possibilità di eliminare i wp, selezionando l'opzione e poi il wp che si vuole eliminare. Per evitare la cancellazione involontaria di altri punti l'opzione si disabilita da solo dopo aver cancellato un wp.

Una volta creato il volo con tutti i way point è possibile simulare il passaggio dell'UAV e la proiezione a terra di tutte le foto prese dal sensore. (Figura 4.8)

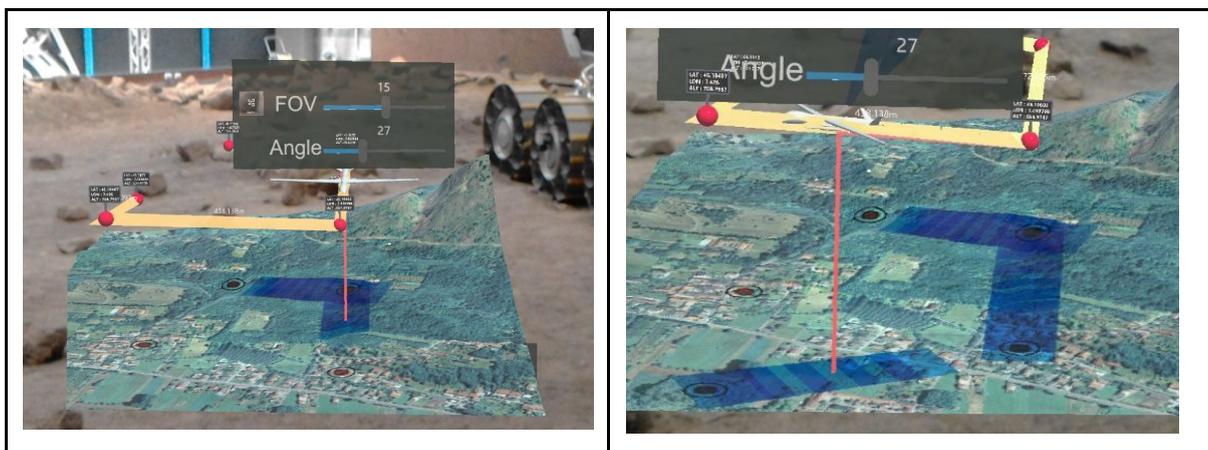


Figura 4.8 sequenza simulazione foto sensore

4.2.4.2 Impostazioni Field Of View

Una volta creato il volo è possibile settare i parametri del Field Of View per la simulazione delle riprese del sensore. Il pannello permette di impostare la dimensione del FOV del sensore, il rapporto di questo, l'angolo e la frequenza di scatto. È possibile valutare le impostazioni mediante la proiezione del FOV sul terreno e/o mediante un pannello che mostra ciò che vede il sensore. (Figura 4.9)

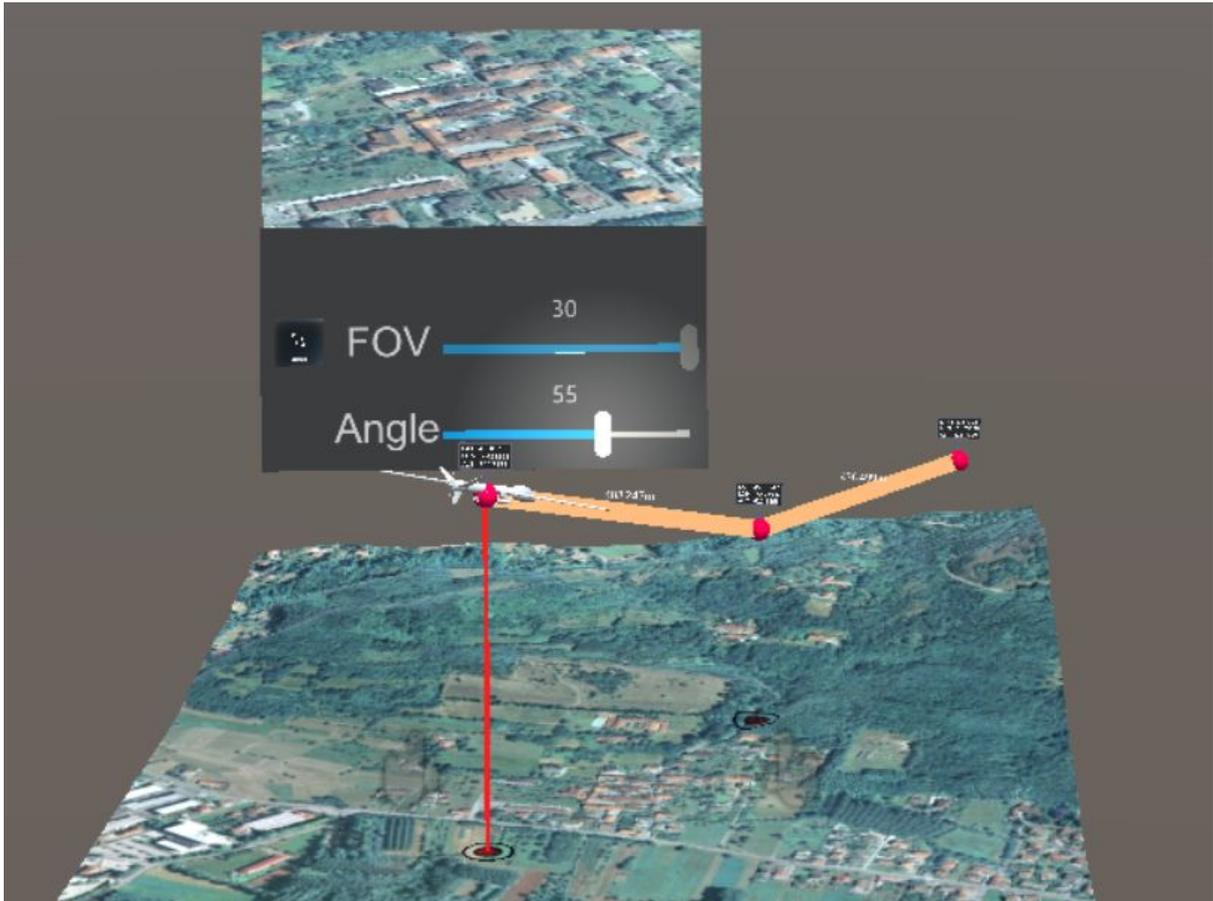


Figura 4.9 viewport camera UAV

Durante il volo il sensore lascia a terra una impronta del FOV con una frequenza pari al valore di rilevamento impostato. Questo permette di valutare l'area di raccolta dati e la sovrapposizione tra le varie riprese aeree utile per le diverse applicazioni. Ad esempio la ricostruzione altimetrica di terreni mediante foto necessita di un'alta percentuale di sovrapposizione delle riprese, a differenza delle riprese aeree per la valutazione di immagini dove si desidera la sovrapposizione strettamente necessaria alla correlazione delle due immagini successive. (Figura 4.8)

4.3 Vista tecnologica

Diagramma sottostante mostra le interrelazioni tra le tecnologie usate per lo sviluppo dell'applicazione (Figura 4.10)

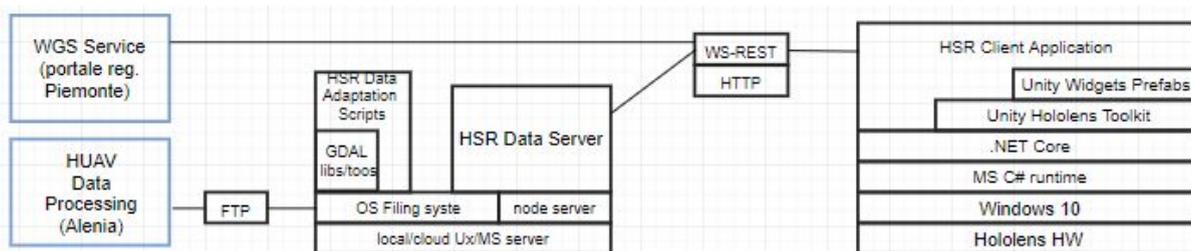


Figura 4.10 vista tecnologica

Come si vede dallo schema, il sistema è composto da:

- Un server locale;
- Un servizio WMS;
- Una applicazione UWP per Hololens

L'applicazione comunica con entrambi i server tramite protocollo HTTP

4.3.1 Server locale

Sviluppato in Javascript ed eseguito su piattaforma Node.js. Funge principalmente da repository per i DTM ma è predisposto per ospitare tutte le informazioni georeferenziate suddivise per territorio (Figura 4.11), come possono essere voli, truppe di terra, elementi generici. Per permettere il funzionamento dell'applicazione in LAN il repository permette di immagazzinare informazioni di texture e overlay.

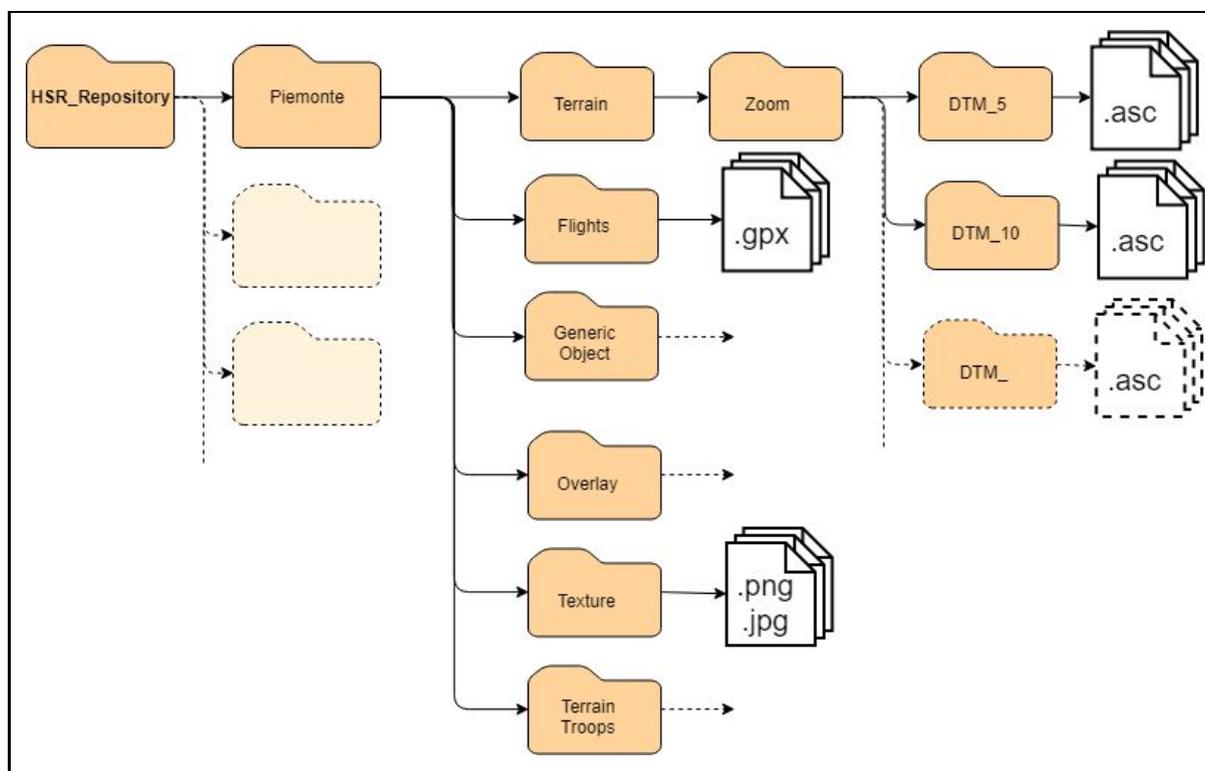


Figura 4.11 struttura repository

Nello sviluppo dell'applicazione l'utilizzo di un server esterno si è visto necessario per una molteplicità di motivi. Il primo riguarda lo spazio di storage dell'Hololens. Per dare un'idea, i DTM a 5 metri del Piemonte, nella loro totalità, sono nell'ordine di grandezza dei 8GB, ancora in formato GeoTiff. Una volta convertiti in ASCII si parla di un aumento intorno al 470%.

L'argomento conversione ci porta al secondo motivo dell'utilizzo di un server. Il processo di conversione da GeoTiff a file ASCII utilizzabili dall'applicazione si basa principalmente sulle librerie di gestione e trasformazioni di file geospaziali raster o vettoriali GDAL. Dopo una serie di prove e consultazioni con specialisti Microsoft si è concluso che le librerie non sono portabili sul framework .NET Core. Tutte le conversioni vengono dunque gestite dal server. Nella fattispecie dal file GeoTiff UTM viene convertito in coordinate WGS84

```
gdalwarp -t_srs '+proj=longlat +datum=WGS84' in.tif output.tif
```

Una volta portato in WGS84 viene trasformato in file ASCII

```
gdal_translate -of AAIGrid output.tif test_grid.asc
```

In fine il valore NULLVALUE del raster viene portato pari a zero e modificato in tutto il file. Quest'ultima modifica potrebbe portare a rappresentazioni incorrette in determinate situazioni, ad esempio se si dovessero rappresentare rilevazioni sotto il livello del mare o se il sensore impostare uno zero arbitrario. Dato l'ambiente di utilizzo di questa prima versione si è valutato che i margini d'errore fossero praticamente nulli e il beneficio nella rappresentazione valesse la probabilità d'errore.

In fine, guardando a possibilità future, l'implementazione di un server risulta pratica in una configurazione con più utenti. In questo caso i dati vengono processati e/o aggiornati una sola volta per tutte le macchine senza creare problemi di versionamenti diversi o mancanza di dati in alcune device.

Questi principi non valgono solo per i terreni, demandando alla struttura repository l'immagazzinamento dei dati il device rimane trasparente all'applicazione. Da qualunque device si lanci l'applicazione si avranno i medesimi dati/risultati.

Una volta elaborati i dati in entrata, il server è pensato per avere esclusivamente la funzione di repository nei confronti dell'applicazione su Hololens. Per mantenere questa filosofia inizialmente alla richiesta di un terreno viene passato all'applicazione l'intero DTM che conteneva la zona richiesta. Date le dimensioni dei file interi, intese come numero di celle di matrice, l'applicazione su Hololens esaurisce la memoria heap disponibile sul device. Si è dunque sviluppato un metodo di cropping. Mediante HTTP request è possibile caricare una TILE, ovvero una porzione, del DTM completo. Le tile hanno dimensione fissa misurata in numero di pixel, intesi come valori di rilevazione nell'originario GeoTiff. Il DTM viene diviso in tile, mediante la query è possibile acquisire i dati della tile interessata indicando l'indice di riga e di colonna. Ogni volta che viene estrapolata una tile dal raster completo le vengono associati dei metadati ri-calcolati dai quelli originali ed arricchiti dalle informazioni di riferimento del nuovo sistema di riferimento virtuale.

4.3.2 Servizio WMS

Come detto precedentemente le immagini applicate come texture al modello tridimensionale del terreno provengono dal servizio WMS della regione Piemonte.

In questo caso l'applicazione, dopo aver recuperato i metadati delle tile, si calcola le quattro coordinate, BottomLeft e TopRight, che servono a definire il bounding box, richiesto dalle specifiche della query per il WMS. Tutti gli altri parametri sono costanti, pre-impostati nel codice.

Sfruttando un servizio WMS si demanda al server la creazione della tile richiesta in termini di risoluzione e dimensione. Per poter fornire immagini a diverse scale vengono prodotte "mattonelle" con soluzioni molto diverse. L'immagine sottostante (Figura 4.12) mostra il concetto di tiling dell'immagine

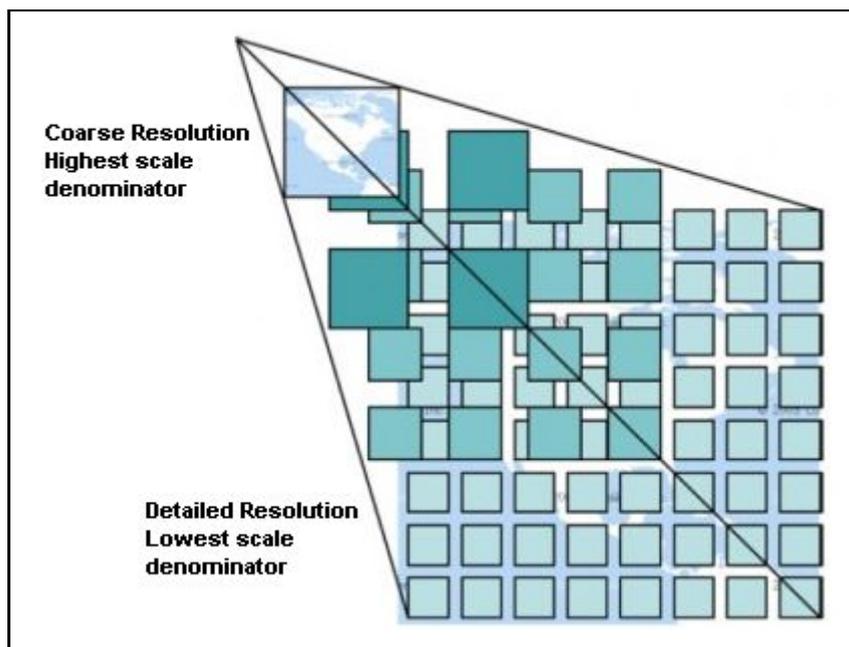


Figura 4.12 tiling dell'immagine

4.3.3 Applicazione UWP per Hololens

L'applicazione HSR è una applicazione Universal Windows Platform (UWP)(Figura 4.13) sviluppata sul game engine Unity con framework .NET Core. Microsoft inoltre mette a disposizione un insieme di script e componenti per lo sviluppo di applicazioni di MR per UWP, specializzate per supportare HoloLens e Windows Mixed Reality Headset chiamato MixedRealityToolkit.

- UWP e .NET Core: UWP è una API sviluppata da Microsoft e introdotta con Windows 10. L'obiettivo di questa piattaforma è di aiutare gli sviluppatori a creare applicazioni adattabili per più piattaforme(Figura 4.13)



Figura 4.13 immagine dal sito ufficiale di Microsoft, piattaforme UWP

Le UWP sono basate su quella che si può considerare una versione multipiattaforma (Windows, Linux, macOS) e cross architecture (x86, x64, ARM) del .NET Framework chiamata .NET Core. È una piattaforma open source che usa la licenza MIT⁹ compatibile con .NET Framework, Xamarin e Mono, attraverso .NET Standard.

- Il Mixed Reality Toolkit per Unity è una raccolta di componenti e script con l'obiettivo di semplificare e accelerare lo sviluppo di applicazioni per HoloLens riducendo le barriere per creare applicazioni in MR e testarle.

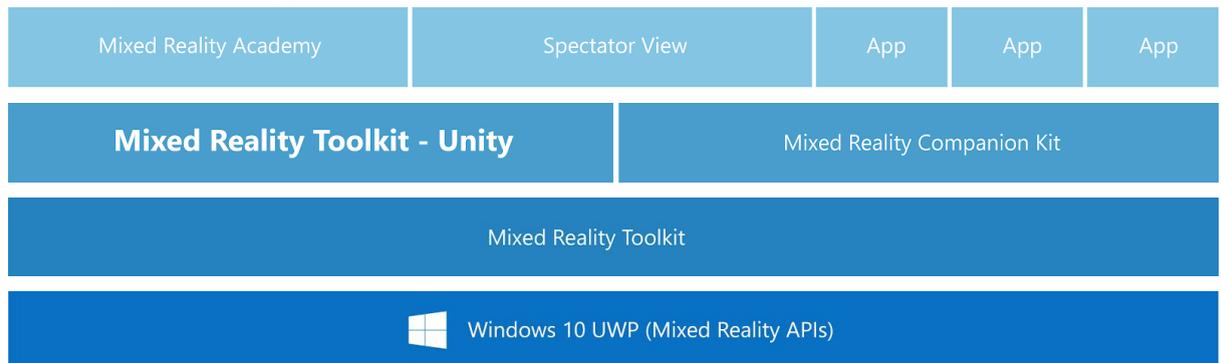


Figura 4.14 immagine dal sito ufficiale di Microsoft, struttura sviluppo MR app

- Data la giovane età di questo progetto subisce continuamente aggiornamenti¹⁰. Uno di questi comprendeva un ristrutturamento totale delle librerie base per la gestione degli input, costringendo ad un revisionamento dell'intera gestione degli input dell'applicazione.

Ad ogni modo il Mixed Reality Toolkit è un progetto open-source con l'obiettivo di condividere una serie di componenti e blocchi funzionali per le interazioni più comuni e per l'interfaccia utente sviluppate per la MR.

⁹ MIT license provides one with the permission to reuse proprietary software, given the condition that the distributors include the copy of original MIT license with the distribution code

¹⁰ solo nel corso di questo progetto ha subito tre aggiornamenti importanti ed una decina di modifiche e bug fix

4.4 Aspetti realizzativi

4.4.1 Terreni

La funzione di visualizzazione dello Scenario Operativo produce la visualizzazione tridimensionale navigabile di una determinata porzione del Terreno di operazioni con la possibilità di selezionare ciascun punto del terreno visualizzato e di determinarne la posizione georeferenziata

La realizzazione di questa prestazione si appoggia principalmente sulla funzione di rendering visuale di Mesh fornita dal motore di Unity/Hololens, ma ha dovuto tenere conto, nella realizzazione, di alcune caratteristiche particolari indotte dalla peculiarità sia tecniche che funzionali dell' oggetto Terreno e della prestazione attesa:

In particolare è stato necessario gestire i seguenti aspetti:

Numero di vertici dell'oggetto "Terreno":

I dati di Elevazione utilizzati per la modellazione del terreno possono essere caratterizzati da una quantità di dati elevata di "vertici" e un ambito territoriale "sufficientemente esteso" (e.g: qualche chilometro quadrato) reso, ad esempio, alla risoluzione di 5 metri dei DTM disponibili, può raggiungere facilmente un numero di vertici ampiamente superiore ai 10^5 , 10^6 unità.

Se da un lato il motore di rendering grafico di Unity su HW grafico anche non specializzato, ha in generale la capacità di rendere con sufficiente efficienza scenari con questo numero di vertici, dalle prove effettuate preliminarmente su Hololens si è rilevato un limite pratico nella efficienza del rendering di singole Mesh sul device per un numero di vertici superiore a $N \cdot 10^4$ vertici per Mesh, con una brusca decrescita intorno ai $6-7 \cdot 10^4$ vertici. Si è quindi optato per la modellazione del Terreno tramite l'unione di insiemi di segmenti di Terreno di dimensione prefissata (Terrain Tiles) in una Terrain Tile Grid che, nel suo complesso, rende lo Scenario Operativo modellato.

Navigabilità laterale dello scenario:

Per ovvie ragioni di limitazione sulle ampiezze delle visualizzazioni olografiche, l'applicazione visualizza, ad ogni dato istante, solo una porzione "delimitata" del Territorio (finestra di visualizzazione) , ma consente all'Utilizzatore, tramite appositi comandi di navigazione, di "muovere" il focus della visualizzazione su parti adiacenti cambiando progressivamente la porzione di Territorio visualizzata.

Allo scopo di consentire una gestione "fluida" della prestazione, si è fatto leva sulla tecnica di Tiling di cui si è parlato al punto precedente, scegliendo opportune dimensioni per la Tile Grid e utilizzando tecniche di caricamento dinamico e, ove possibile, proattivo delle informazioni necessarie alla funzione.

Complessivamente il terreno visualizzato viene quindi ottenuto tramite una insieme di "Terrain Tiles" elementari indipendenti opportunamente disposte a griglia. L'effetto

complessivo “unitario” del rendering viene coordinato da un apposito oggetto “Terrain Tile Grid” (Figura 4.14) che provvede al reperimento delle Tiles elementari necessarie allo Scenario in oggetto, alla loro disposizione spaziale relativa e alla realizzazione degli elementi di raccordo necessari a colmare i “gap” tra le Tiles indipendenti (i Glue Tapes).

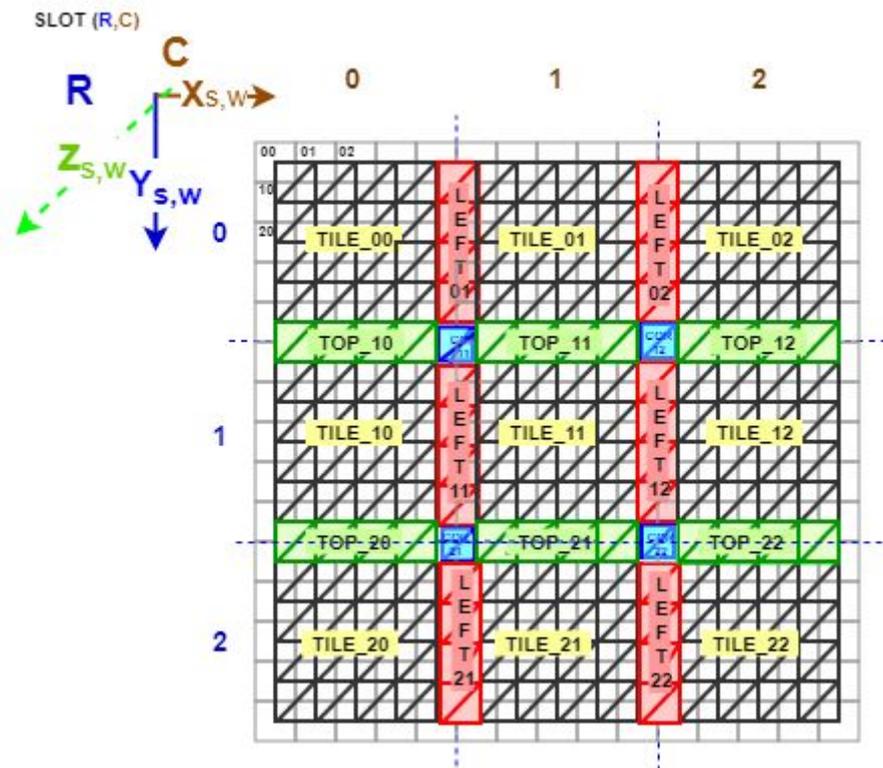


Figura 4.14 TileGrid

La creazione delle “Glue Tapes” deriva dalla creazione di una “Tile Mesh”, ovvero dal passaggio da informazione del raster a mesh per il motore di rendering. In questo passaggio la definizione dei triangoli, elemento primitivo della mesh, perde l’informazione dei vertici di bordo mesh per via del cropping del raster completo. A causa di ciò le Tile adiacenti risultano separate da un buco unitario. Per non modificare la logica delle Tile in base alla posizione, si è creato un oggetto “Glue Tapes” che prende l’informazione dall’oggetto “Terrain Tile Grid”, la cui struttura rimane costante. I Glue Tapes costruiscono una mesh prendendo i vertici delle Tile adiacenti ed “unendole”. I Glue Tapes si dividono in tre tipologie, left, top e corner come si può vedere nell’immagine soprastante.

La figura seguente evidenzia un esempio di Tile Grid con le componenti Tile elementari e gli elementi di raccordo tra queste. (Figura 4.15)

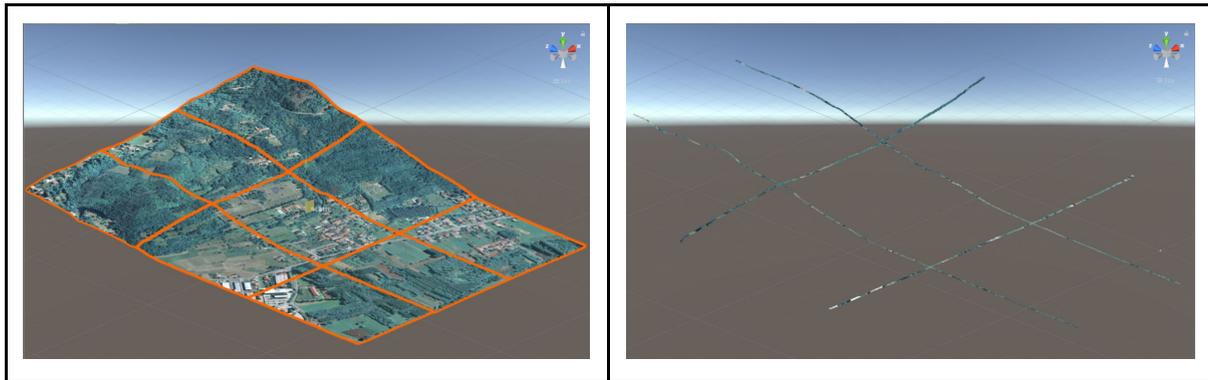


Figura 4.15 Tiles (a sinistra) e Glue Tapes (a destra)

Il seguente diagramma (Figura 4.16) illustra i principali passaggi di elaborazione effettuati per la preparazione del rendering di una singola Tile, e le elaborazioni aggiuntive per l'inserimento della Tile nella Grid complessiva.

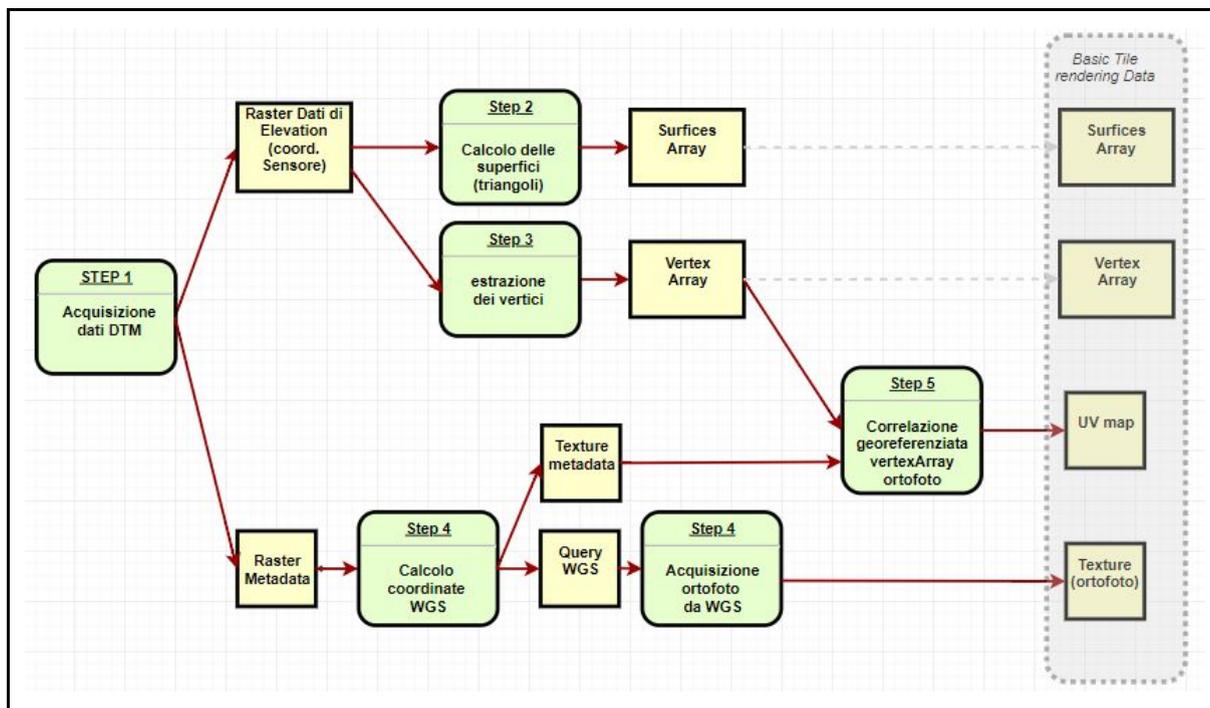


Figura 4.16 flusso elaborazione per Tile

Si può notare dal grafico come, partendo dallo stesso DTM, vengono creati due branch separati per la creazione della mesh e per l'applicazione della texture. Questo è stato reso necessario a causa della provenienza delle informazioni da due sorgenti diverse ma, allo stesso tempo, rende più flessibile il sistema. È infatti possibile, in un secondo momento, applicare ulteriori texture proveniente da diverse fonti purchè georeferenziate.

Un'altro vantaggio è la possibilità di parallelizzare le operazioni ottimizzando i tempi di risposta del sistema. In questa versione del software non è stato possibile a causa di un bug nelle del componente UnityEngine.Networking, che verranno affrontate nello specifico nel paragrafo "Valutazioni". Si veda dunque come sono avvengono attualmente le interazioni.

Il seguente diagramma (Figura 4.17) fornisce una prospettiva temporale delle interazioni che intercorrono tra i vari sottocomponenti della applicazione nel processo di visualizzazione dello Scenario Operativo.

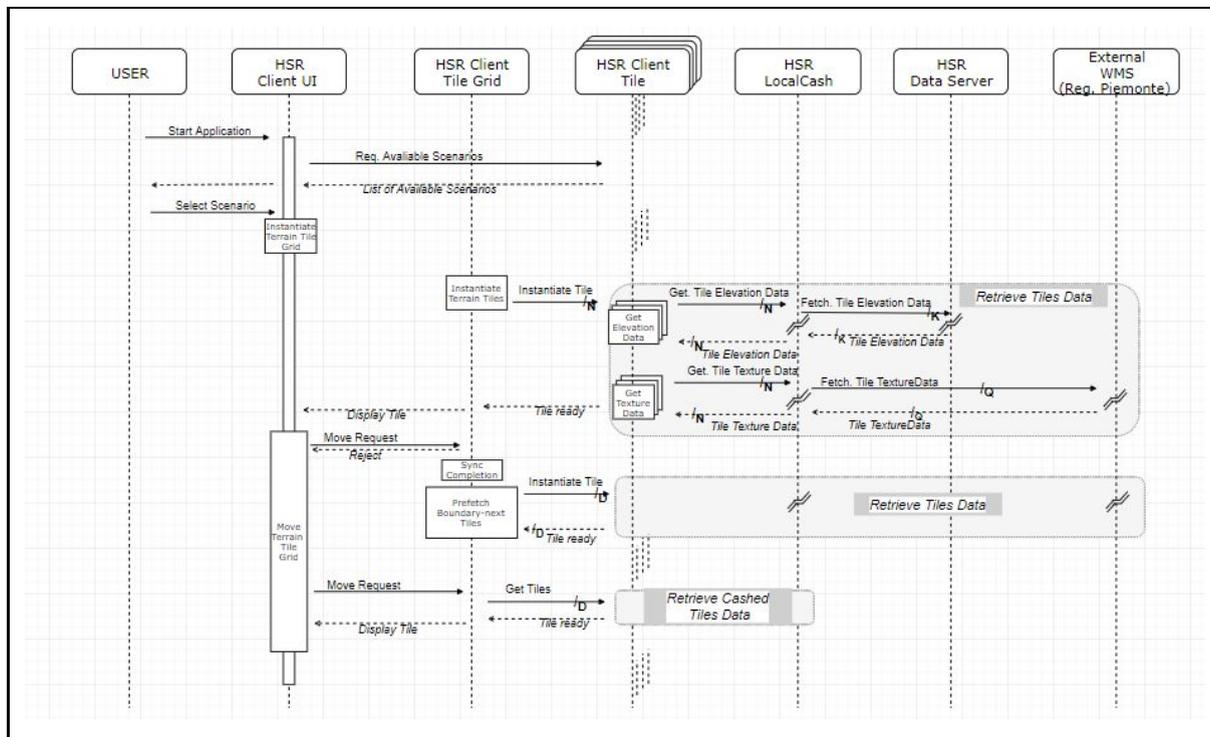


Figura 4.17 UML sequence diagram creazione terreno

Il diagramma evidenzia in particolare:

- Il caricamento indipendente, potenzialmente parallelo, dei dati necessari al rendering delle singole Tiles
- l'utilizzo di una cash per i dati delle singole tiles, a supporto del riuso dei dati acquisiti in caso di riuso delle Tiles in "finestrature" successive dello Scenario Operativo.
- l'utilizzo di pre-fetching delle Tiles del primo perimetro esterno, i.e: le Tiles candidate a essere incluse in una delle possibili 4 finestrature successive dello Scenario Operativo
- l'utilizzo di Tiles direttamente reperite dalla cash, a pre-fetching completato, con conseguenti tempi di reazione ridotti

Il diagramma seguente (Figura 4.18) fornisce una vista della struttura di inheritance e collaboration fra le principali entità coinvolte nella realizzazione della funzione di Visualizzazione della Grid di Terreno dello Scenario Operativo.

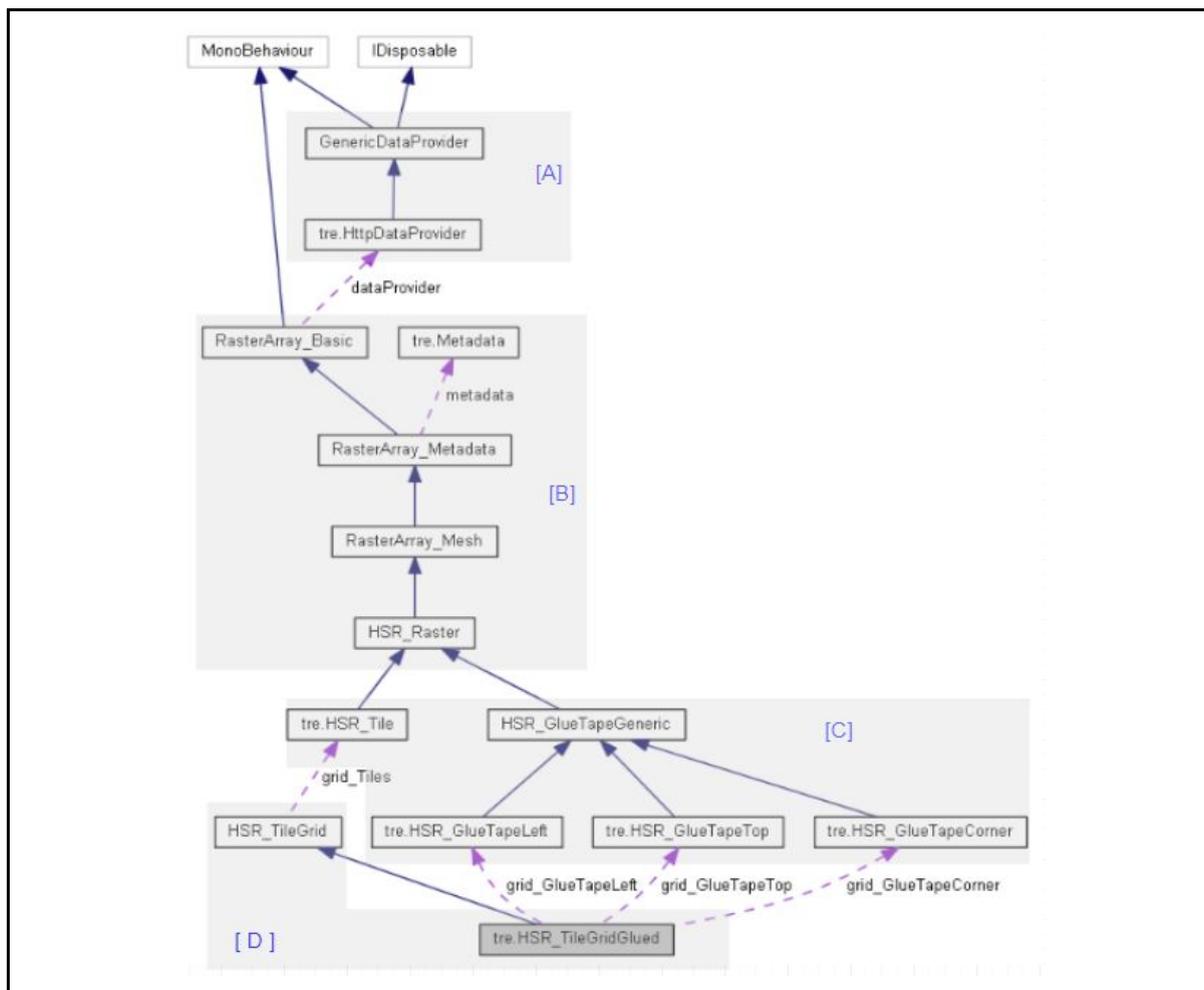


Figura 4.18 UML struttura di inheritance

Il diagramma evidenzia:

- [A] l'utilizzo di un DataProvider generalizzato per la acquisizione dei dati elementari (caratterizzato da una Interface basata sulle Signatures elementari/generalizzate hasNext() e next()), specializzato in un DataProvider su HTTP¹¹.
- [B] la specializzazione progressiva dell'oggetto Raster di dati a partire dalle funzioni base di memorizzazione e accesso dei dati del RasterArray_basic, a seguire con le estensioni relative alla gestione dei Metadati associati al raster e infine con le funzioni di generazione delle informazioni necessarie al rendering grafico dell'oggetto, a partire dai dati del raster e dei metadati, (precedentemente illustrato nel dataFlow diagram xxxx)
- [C] la specializzazione, e riutilizzo, dell'oggetto Raster sia come entità Tile che come entità GlueTape (ulteriormente specializzate poi nelle tre versioni di oggetto raccordo "laterale"/"superiore"/"diAngolo".
- [D] la specializzazione dell'oggetto TileGrid elementare, preposto alla gestione delle sole Tiles nell'oggetto TileGridGlued che estende le funzioni di TileGrid con la gestione dei raccordi grafici e la resa unitaria del Scenario Operativo.

¹¹ Il componente specializzato incapsula e maschera l'interazione HTTP e rende accessibili i dati tramite i metodi generalizzati della Interfaccia DataProvider.

Sono stati effettuati una serie di test per l'ottimizzazione empirica del sistema di generazione dei terreni per numero di vertici a Tile e numero di Tile per Grid (Figura 4.19)

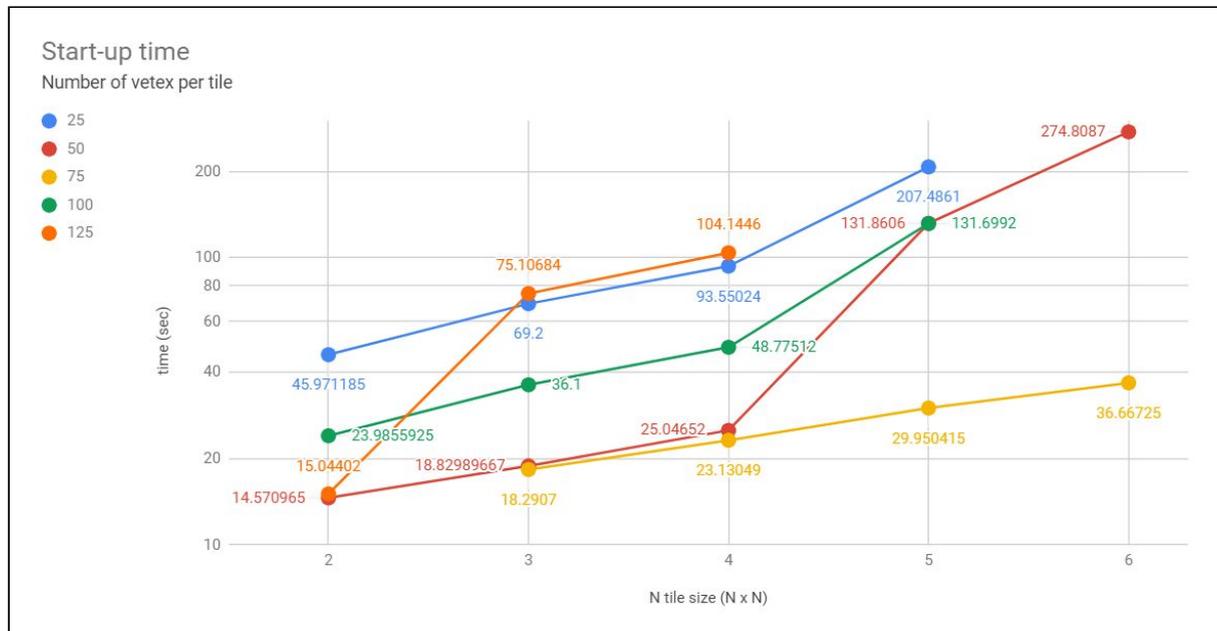


Figura 4.19 grafico risultato test startup time

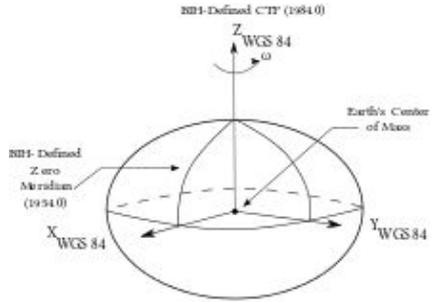
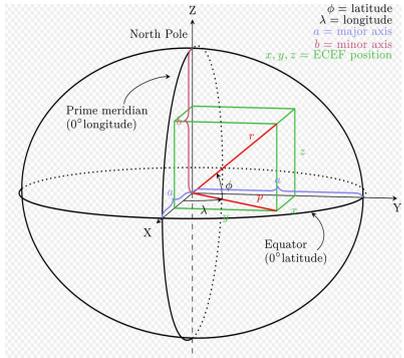
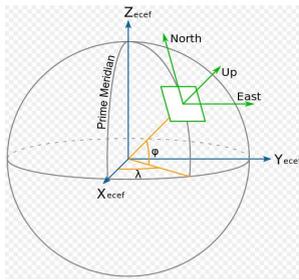
Si è riscontrato il miglior equilibrio tra porzione di terreno mostrata e tempi di accesso nella configurazione con 75x75 vertici per Tile e 3x3 Tile per Grid. Questo consente, con i DTM forniti, di visualizzare delle porzioni di terreno di 1125 metri con una risoluzione di texture pari 1.09 metri per pixel.

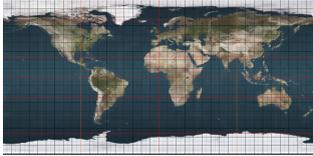
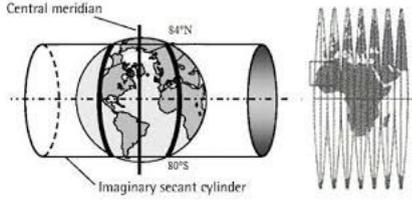
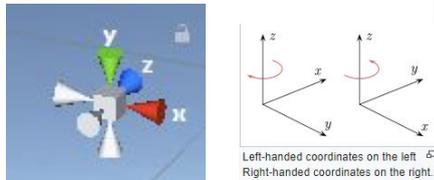
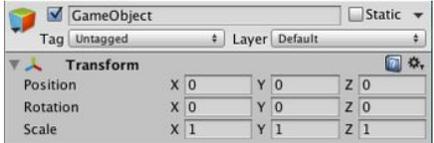
4.4.2 Sistemi di riferimento

I dati geospaziali subiscono svariate trasformazioni di coordinate passando dai sistemi di riferimento cartografici a quelli della rappresentazione in Mixed Reality sollevando tutta una serie di problematiche, dalla conversione ed errori connessi ai rapporti proporzionali, alle approssimazioni e limiti di rappresentazione.

Nella tabella sottostante possiamo vedere tutti i sistemi di coordinate che sono stati presi in considerazione.

Acronimo HSR	Descrizione	Immagine
GEO (GeoRef) "assoluto" (WGS84)	Origine: meridiano Greenwich-equatore su (un) geoide Coordinate: - latitudine ("verticale", "y") in gradi decimali , +/- 90° - longitudine ("orizzontale", "x") in gradi decimali +/- 180°,	

	- elevation "radiale" in metri	
	<p>WGS84</p> <p>centro: nel centro di massa della Terra; asse Z: passante per il polo Nord asse X: scelto in modo che il meridiano di Greenwich giaccia sul piano XZ asse Y: scelto in modo da dare una terna destrorsa, ovvero tale che un osservatore posto lungo l'asse Z veda l'asse X sovrapporsi a Y con moto antiorario, il che pone Y in Asia.</p>	 <p>Figure 1.1 WGS 84 Reference Frame</p>
GEM (GeoRef in metri) anche	<p>Come GEO, con y (latitudine) e x(longitudine) in metri</p> <p>$x_{GEM} = 2\pi * \text{Radius} / 360 * x_{GEO}$ $y_{GEM} = 2\pi * \text{Radius} / 360 * y_{GEO}$ $z = z$</p>	
GEOxyz "relativo"	<p>Come GEO , ma "Relativo" ad una altro punto di riferimento xyz (utile per aumentare la precisione o per le trasformazioni)</p>	
ECF (Earth Cartesian Fixed) "assoluto" (ECEF) (WorldGlobePosition) WGP (World Globe Position) GWP (Geo World Position)	<p>Origine: centro gravitazionale terrestre Coordinate (terna cartesiana Right-Handed, in metri) :</p> <ul style="list-style-type: none"> - x ortogonale all'equatore (passa per 0° X equatore) - y ortogonale all'equatore (passa per +90° X equatore)? - z Asse di rivoluzione terrestre verso Nord 	 <p>ϕ = latitude λ = longitude a = major axis b = minor axis x, y, z = ECEF position</p>
ENU (East, North, Up) "relativo"	<p>Origine: punto arbitrario sulla superficie Coordinate (terna cartesiana Right-Handed , in metri) :</p> <ul style="list-style-type: none"> - x tangente al parallelo, in metri, verso est - y tangente al meridiano, in metri , verso nord - z sulla verticale (raggio) in metri. 	

<p>MER (mercator projection) "assoluto"</p>	<p>Proiezione di mercatore (quale ? ... una....) Origine: meridiano Greenwich-equatore Coordinate (terna cartesiana in metri) : - x lungo l'equatore - y meridiano greenwich - z elevation (non si usa)</p>	
<p>UTM (Universal Transverse of Mercator)</p>	<p>Una delle proiezioni di Mercatore: Valido da 80° nord a 80° SUD. Solo per quadri "locali" 60 fusi di 6° long, 20 fasce 8° lat => 1200 zone Origine: Greenwich + 500000m</p>	
<p>SEN (sensor) "relativo"</p>	<p>Origine Lettura: top-left del sensore Raster con x,y interi e z(elevation) float Metadati per avere la GeoRef e la cellsize (origine solitamente LL LowerLeft) double ENU con z "rovesciata" (z "cresce" verso il drone)</p>	
<p>MOD</p>	<p>Origin: "Unity Model" (0,0,0) Cartesian "Left-Handed" → Unity "Right-Handed" - x orizzontale, "pixels"/"cells" - y verticale. "pixels"/"cells" -z orizzontale (rovesciato! ... o rovesciato "y"), "pixels"/"cells" Come UnityModel con 1 unità <==> 1 metri</p>	
<p>MIX</p>	<p>HololensMixedReality Transform applicate al MOD in Unity Applicate allo start per posizionare e scalare il Terreno Applicate dall'utente durante l'utilizzo</p>	

GEO

È Il sistema di coordinate geografiche in cui vengono acquisiti i dati in origine.

Data la molteplicità delle rappresentazioni geografiche e, conseguentemente, la molteplicità di possibilità di espressione dei dati con sistemi diversi, si è dovuto prenderne uno di riferimento.

Si è deciso di usare il WGS84 (sigla di World Geodetic System 1984), sistema geografiche geodetico, basato su ellissoide, usato dai sistemi satellitari per la navigazione GPS. La decisione è supportata inoltre dalla necessità di usare il WGS84 per le query al WMS.

Per rispettare questo sistema di coordinate i DTM che vengono inseriti nel repository (solitamente con un sistema di coordinate UTM) vengono convertiti in WGS84. La differenza più significativa tra i due sistemi è che nell'acquisizione dati il sistema UTM prevede delle distanze tra i pixel uguali tra l'asse x e quello y. Mentre il WGS84 viene affetto dalla rappresentazione ellissoidale dunque la y risulta più grande della x, distorcendo leggermente l'immagine come si può vedere dal confronto delle due immagini sottostanti (Figura 4.20)

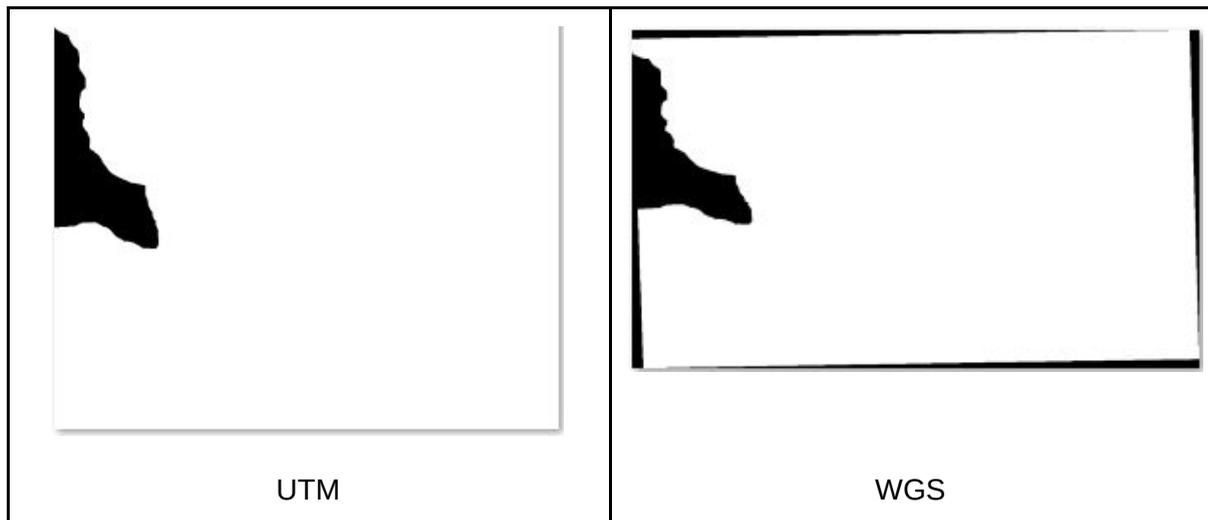


Figura 4.20 confronto fra GeoTIFF UTM e la sua conversione in WGS84

I metadati espressi in WGS84 vengono elaborati poi da Unity, bisogna dunque fare un paio di considerazioni sulla gestione del sistema di coordinate di Unity.

La matematica del sistema di coordinate in Unity è basata su 32bit float che causa dei problemi di imprecisione dati dall'arrotondamento quando si lavora con le dimensioni originali. Arrivano ad arrotondamenti alla quinta cifra decimale pari a circa un metro¹². Per sopperire a questo problema i metadati vengono gestiti mediante variabili double da 64bit.

Il secondo problema di precisione è portato dallo Unity Space World. Come conversione da WGS84 viene usata la "Proiezione di Mercatore EPSG:3857". Una volta che si assegnano i valori alle Transform dei GameObject di Unity questi vengono portati ad una risoluzione di 32bit. Per limitare questa imprecisione si cerca di lavorare con distanze invece che con posizioni assolute. Si salva dunque l'offset in una variabile double mentre il delta risulta un numero più piccolo e dunque con più precisione anche se float. Allo stesso modo si cerca di posizionare i GameObject il più vicino possibile all'origine. In questo modo vengono usati più bit per la risoluzione decimale aumentando la precisione. In sostanza la posizione relativa riduce la perdita di precisione.

SEN

Con SEN ci si riferisce alla rappresentazione di una porzione di territorio mediante immagini GeoTiff, ovvero alla matrice del raster delle altezze rilevate. È da questa matrice che verrà costruita la mesh del terreno. Per poter trasportare i dati su un modello tridimensionale in Unity bisogna fondamentalmente passare da un sistema "Right-handed" ad uno "Left-handed". Gli effetti di questa differenza sono che la figura viene letta con x che cresce da sinistra a destra e y dall'alto verso il basso, in Unity (per creare una superficie piana) la x cresce sempre da destra a sinistra mentre bisogna sviluppare la superficie sull'asse z che cresce dal basso verso l'alto (Figura 4.21):

¹² Questo valore è puramente indicativo. L'errore è maggiore in latitudine rispetto alla longitudine ed aumenta in generale allontanandosi dall'equatore rendendo fortemente variabile.

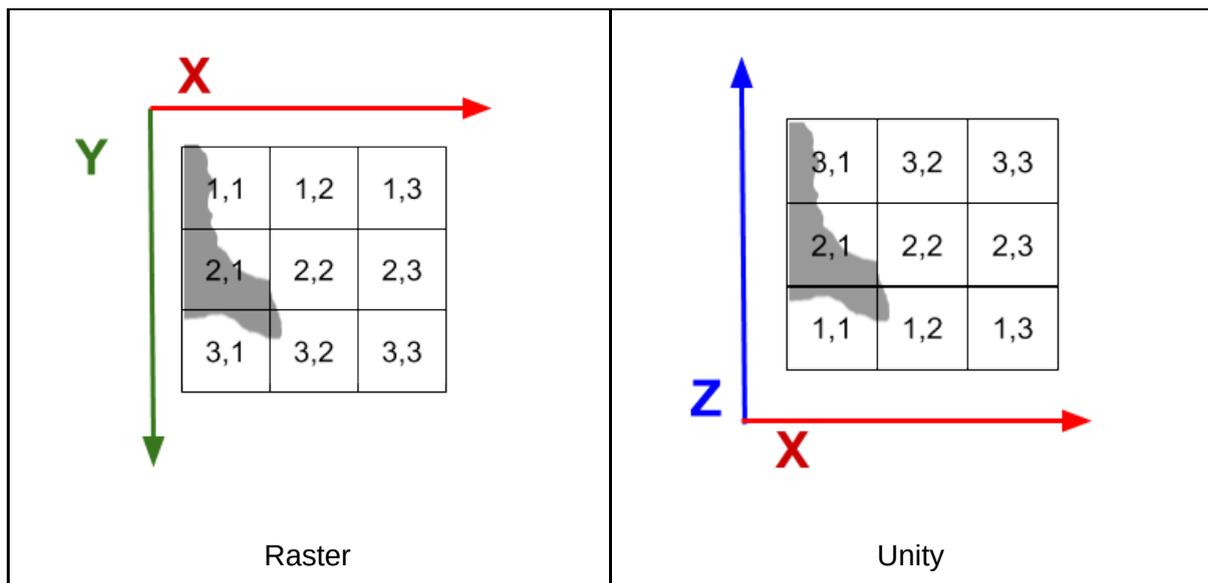


Figura 4.21 confronto coordinate da raster a Unity

Bisogna dunque copiare la matrice bidimensionale del raster in una matrice con gli indici di linea pari alla dimensione della matrice stessa meno il numero di linea ($[0,0] \rightarrow [ySize-0, 0] \dots [1,0] \rightarrow [ySize-1, 0] \dots [N,0] \rightarrow [ySize-N, 0]$).

La stessa cosa avviene per le Tile rispetto alla Grid.

MOD

Con MOD ci si riferisce al passaggio da raster a modello in Unity. L'origine del terreno viene impostata uguale all'origine degli assi di Unity. La distanza tra i pixel viene convertita in una unità di Unity. L'altezza, ancora espressa in metri, viene proporzionata alle dimensioni in x e z usando sempre la proiezione di Mercatore per calcolare il rapporto.

MIX

In fine il MIX si riferisce al modello di Unity trasportato nella mixed reality. Il modello viene scalato per essere fruibile in una stanza. Secondo le indicazioni di Microsoft una unità di Unity corrisponde all'incirca ad un metro nella realtà fisica. Il modello viene dunque scalato per essere una superficie quadrata di due metri e mezzo.

Per lo stesso principio il modello viene traslato affinché, all'avvio dell'applicazione, si presenti centrato sull'utente ad un metro e mezzo di distanza. Questa posizione può essere temporanea poiché è l'unica trasformazione che può essere modificata dall'utente final.

Il diagramma seguente (Figura 4.22) riporta i passaggi di elaborazione per la trasformazione da coordinate SEN a coordinate MOD (con i riferimenti alle Classi del codice HSR dove sono implementati) .

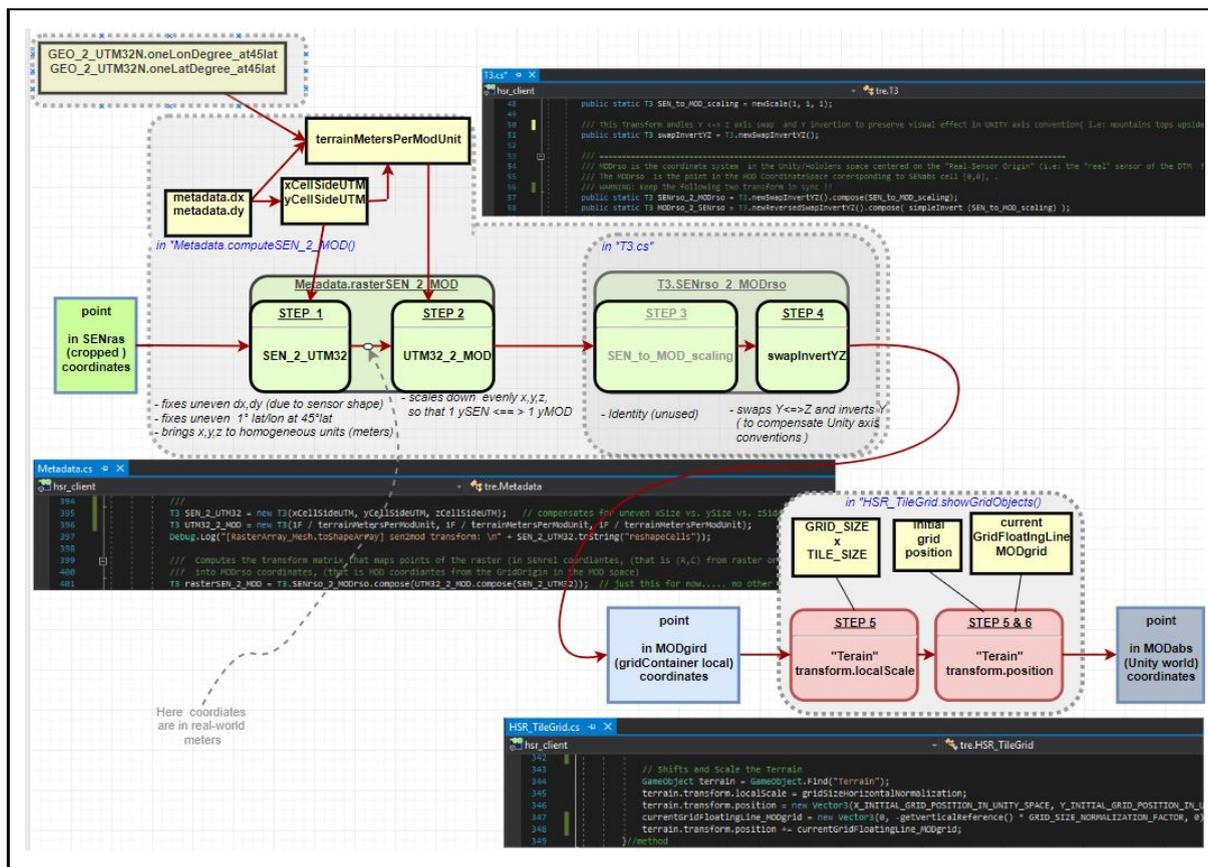


Figura 4.22 passaggi trasformazione coordinate

4.4.3 UX/UI

La mixed reality ridefinisce la concezione di interfaccia utente. Con la MR l'interfaccia utente non è più mediata da uno schermo piatto ma diventa un tutt'uno con l'ambiente circostante, reagendo in funzione a dove si posa lo sguardo e da come si interagisce gestualmente con esso. La UI acquisisce proprietà 3D e fisiche come posizione, distanza, peso e rigidità. Un'altra rottura con il passato viene dall'interazione con l'interfaccia. Da sempre l'interazione uomo-macchina è stata mediata dall'hardware come tastiera, mouse, touchscreen, ecc.. Una Interfaccia 3D che interagisce con l'environment risulterà in una totale ridefinizione dell'interazione e abbandono delle vecchie pratiche, o quasi. Val Head, Design Evangelist, presso UX Innovation di Adobe, ritiene che la MR racchiuderà le caratteristiche migliori di entrambe le interfacce, 2D e 3D. Lei dice "The mixed reality user interfaces of the future will draw upon influences from both old and new techniques....we can see how 2D elements of human-computer interactions can be ported into a 3D environment and still be successful and intuitive to users." aggiungendo anche "Although animation that is used in a 3D spatial canvas will be different than it is on 2D screens, they have the same intentions — to show feedback to users and help them to keep track of where to go. But when it comes to moving objects, we will have to be more accurate for physicality, weight, space and timing. It should be more realistic because interactions will be occurring in real space and time." Si va dunque a ridefinire non solo la UI ma anche la UX, ovvero la User Experience.

A tale proposito Microsoft, basandosi sull'esperienza del loro Sr. Holographic Designer, Marcus Ghaly, fornisce tre suggerimenti per la progettazione di UI e UX di un'applicazione per HoloLens.

1. Le persone tendono a non girare intorno al soggetto su cui stanno interagendo.

È stato testato che se si costruiscono le superfici di lavoro olografiche circolari invece che rettangolari le persone sono più portate ad usufruire pienamente dello spazio di lavoro. Questo esempio fa capire come bisogna cercare di far sentire a proprio agio l'utente finale a sfruttare tutto lo spazio fisico, valore aggiunto di HoloLens

2. Può succedere che la finestra di dialogo sia fuori dal campo visivo olografico.

Ogni tanto l'utente potrebbe guardare in direzioni differenti rispetto a quella dell'oggetto che richiede la sua attenzione. Su un computer solitamente si fa comparire una finestra di dialogo ma su un ambiente 3D risulterebbe occludere la visuale. Dopo diversi esperimenti si è trovato che il modo più efficace per attirare l'attenzione dell'utente in un ambiente 3D sia quello di usare dei spatial sound, raggi luminosi o, forse il più efficace, l'effetto sentiero puntinato (come le nuvolette dei pensieri nei fumetti) con i puntini che vanno rimpicciolendosi mentre ci si avvicina all'obiettivo.

3. A volte la UI può essere bloccata da altri ologrammi.

Ci sono volte in cui l'utente vuole interagire con l'interfaccia di un ologramma ma questa viene bloccata da un ologramma che risulta più vicino (Figura 4.23). Provando ad avvicinare l'interfaccia utente questa non veniva più coperta ma risultava poco confortevole per il suddetto utente ritrovandosi con l'ologramma soggetto dell'azione lontano e l'interfaccia vicina. Alla fine hanno optato per un effetto detto ghosting della UI.

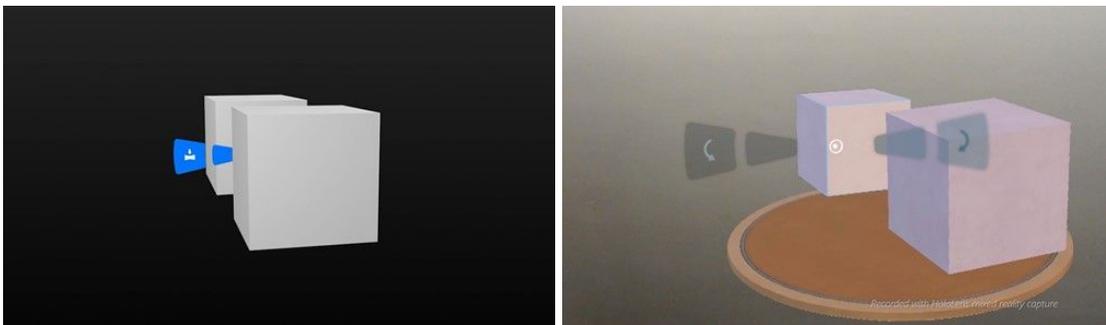


Figura 4.23 immagine da sito ufficiale Microsoft, differenza tra interfaccia utente occlusa (a sinistra) e non occlusa (a destra)

In sostanza bisogna sempre cercare una soluzione per cui l'utente sia a suo agio ad usare i controlli dell'UI anche se questi sono coperti, in modo che il soggetto riesca a completare il suo compito da qualunque punto stia interagendo con l'ologramma

In un'ottica più generale bisogna cercare di progettare l'interfaccia utente tenendo a mente tutta una serie di funzionalità e problematiche nuove.

Priva di analizzare le scelte fatte nella UI di questa applicazione bisogna familiarizzare con due termini inseriti da Microsoft per descrivere il comportamento di una finestra:

- **Billboarding:** con questo termine viene descritto il comportamento di un ologramma che si orienta costantemente verso l'utente. Se l'oggetto è libero di muoversi rimarrà sempre centrato sulla visuale dell'utente rischiando anche di occludere la visuale;
- **Tag-Along:** anch'esso descrive il comportamento di un ologramma e può essere aggiunto ad un oggetto che include il billboarding per l'orientamento ma non per la posizione. Questa interazione risulta essere più naturale infatti l'ologramma cercherà di rimanere sempre nel campo visivo dell'utente ma non sarà bloccata al centro dello sguardo permettendo così di poter visualizzare il resto dell'ambiente risultando meno "aggressiva"

Tenendo a mente i principi di Marcus Ghaly sopra descritti i vari menu dell'applicazione sono sostanzialmente divisi in quattro macrocategorie:

- un menu principale legato al terreno;
- dei menu a finestra liberamente posizionabili per personalizzare lo spazio di lavoro;
- dei menu di "lavoro" che seguono lo sguardo dell'utente per essere sempre a portata di mano;
- dei menu o pannelli contestuali, legati agli oggetti, per l'arricchimento informativo di quest'ultimi ;

Tutto l'insieme è pensata come struttura ad albero con il punto di partenza nel menu principale. Per la facile individuazione di questo menu, soprattutto nell'eventualità di sviluppi futuri con multi-utenti, è stato pensato inizialmente statico, legato al terreno. Detto questo si è pensata anche una funzione più adatta ad un singolo utente e che sfrutti la spazialità data da questa nuova tecnologia.

Rifacendosi al concetto 1) di Microsoft nel progettare la UX, per non legare l'utente ad un utilizzo principalmente frontale è stata prevista una funzione chiamata "tool belt" ovvero "cintura di attrezzi". Questa opzione permette di sganciare l'insieme dei pannelli principali e di legarli all'utente in modo che li abbia sempre a portata di mano. Seguendo l'idea del billboarding ma mantenendolo ad una altezza costante in modo che l'utente ne abbia accesso solo guardando verso il basso ma non intralciare la vista del terreno, dando per scontato che l'utente non entrerà mai fisicamente nel terreno. Ancora differente dal billboarding l'insieme dei pannelli mantiene una posizione parallela con il terreno in modo che il pannello centrale sia sempre rivolto verso l'area di lavoro. Così facendo rende inoltre accessibili i pannelli laterali che se no tenderebbe a ruotare.

Il pannello centrale è dedicato alla gestione del terreno, sulla sinistra un pannello che visualizza l'ora locale ed uno sulla destra per l'accesso alle varie funzionalità (Figura 4.24).



Figura 4.24 pannello centrale dell'interfaccia grafica

Il pannello per la gestione del terreno viene tenuto centrale in modo da facilitare la visione del terreno mentre si agisce sui suoi parametri. Nella fattispecie il pannello presenta le coordinate dell'angolo in basso a sinistra (seguendo la struttura dei metadati del DEM) della porzione di terreno visualizzata, le frecce per visualizzare porzioni diverse del DEM mostrando una tile nella direzione della freccia selezionata per ogni click. Sopra il pannello è posizionato uno slider che funge da moltiplicatore del rapporto tra l'altezza e le dimensioni sul piano per poter accentuare o diminuire le asperità del terreno. In fine vi è la predisposizione per i tasti di zoom in e zoom out (funzione non implementata). Gli spostamenti del terreno possono essere richiamati anche tramite comando vocale, per questo motivo e per lasciare più spazio al terreno si è preferito optare per un controllo concentrato in un posto unico rispetto alle più comuni quattro frecce nei quattro lati della mappa. Quest'ultima opzione permette di una miglior visibilità mentre si muove il terreno ma per essere perennemente accessibile il terreno deve essere ridotto sia nello spazio che nelle altezze.

Come detto precedentemente il pannello di destra dà accesso ad altre funzionalità. La prima di queste è la visualizzazione del menu principale per la visualizzazione dei dati associati al terreno. Il menu ha due opzioni ognuna delle quali apre a sua volta un nuovo pannello (Figura 4.15).



Figura 4.25 menu

Una opzione serve a visualizzare i vari layer del terreno (es. Heightmap, slope map, layer degli incendi) con varie funzionalità che verranno approfondite in un capitolo successivo. Il pannello aggiunge e/o toglie controlli aggiuntivi in base al layer selezionato adattandosi dinamicamente alle esigenze di ciò che si visualizza.

L'altra opzione serve a visualizzare l'elenco dei voli registrati, ognuno dei voli ha a sua volta una serie di opzioni che verranno affrontate in un altro capitolo. Tutto l'insieme di questi menu (Figura 4.26) fanno parte della seconda macrocategoria e sono stati pensati con la medesima filosofia per quanto riguarda la UX.



Figura 4.26 insieme dei pannelli

Ogni menu viene visualizzato mediante un toggle button di un menu più alto nella struttura ad albero che, se deselezionato, fa scomparire il suddetto pannello. Quando un menu/pannello viene attivato compare in fronte all'utente e continua a seguirne lo sguardo rimanendo perennemente al centro dell'attenzione. Come prima operazione bisogna posizionarlo nello spazio mediante la funzione Tap To Place ovvero facendo un gesto di tap nella posizione in cui si vuole che il pannello si fermi. Da questo momento si potrà spostarlo selezionandolo nuovamente in modo che si sganci dalla posizione corrente e metterlo nella nuova posizione desiderata. Qui si è scelto di mantenere la semantica comune alle finestre sia di Windows che di OS ovvero rendendo sensibile a questa funzione solo il bordo alto della finestra dove vi è il suo nome (zona nel riquadro verde nella Figura 4.27)

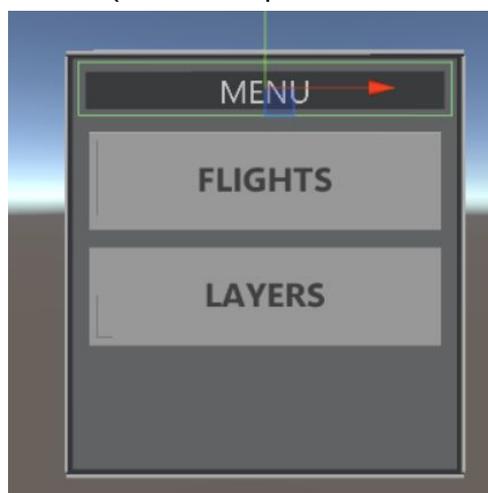


Figura 4.27 bordo sensibile per TapToPlace

Non mettendo un bottone per chiudere il pannello sul pannello stesso ma delegando al menu "padre" questa funzione risolve il problema riscontrato in alcuni casi di un ologramma "incastrato" nel muro. Ovvero può accadere che l'HoloLens finisca la ricostruzione dell'ambiente circostante quando l'ologramma è già posizionato valutando che quest'ultimo sia occluso da un oggetto fisico perdendone così il controllo.

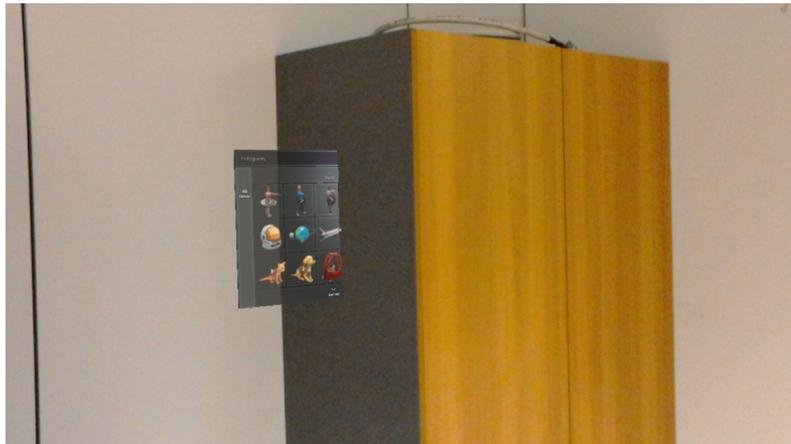


Figura 4.28 esempio pratico di interfaccia “incastrata” nell’ambiente reale. Non è stato possibile interagire con lei in nessun modo

La seconda funzionalità presente sul pannello principale riguarda la creazione e valutazione di nuovi voli. In questo caso viene aperto un pannello che permette di creare ed editare dei way point (Figura 4.29). In questo caso l’ologramma è gestito come un tag-along ma anche qui si è optato per un approccio insolito rispetto le indicazioni generali.

A differenza del punto 3) dei principi di Marcus Ghaly il pannello rimane sempre dietro il terreno. In tal modo rimane sempre a vista e non si rischia di doverlo cercare distogliendo l’attenzione dalle operazioni che si stanno effettuando ma, allo stesso tempo, non interferisce con l’interazione con il terreno (Figura 4.30).

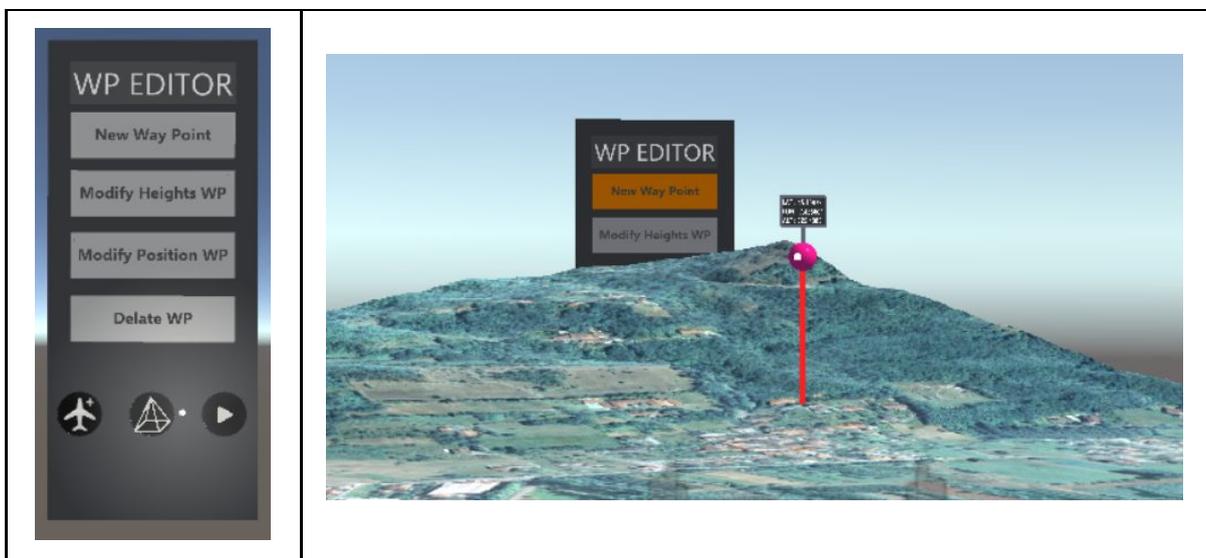


Figura 4.29 way point editor

Figura 4.30 interazione tra pannello volutamente occluso dal terrenp

Le varie funzionalità verranno specificate in un capitolo successivo ma, per quanto riguarda l’esperienza utente è importante vedere come viene aggiunto un way point. L’operazione avviene in due passi. Selezionando un punto sul terreno vengono registrare latitudine e longitudine creando un way point con georeferenziazione pari a quella del punto sul terreno selezionato. A questo punto il way point può essere spostato solo sull’asse verticale per scegliere l’altitudine. Entrambe le operazioni le operazioni vengono effettuate mediante lo

sguardo, o gaze, dell'operatore e non mediante il tracking della mano. Inizialmente si era optato per il tracking, soprattutto per quanto riguarda il settaggio dell'altimetria, ma sono state riscontrate alcune difficoltà per l'operatore. Anche se l'interazione risultava più intuitiva, si è riscontrata una imprecisione maggiore con la mano. A lungo andare il braccio tendeva a stancarsi, portando ad una esperienza utente frustrante, per i continui aggiustamenti, e faticosa. Un altro fattore che ha inciso sulla decisione della tipologia di interazione è il campo di riconoscimento dei gesti, detto gesture frame, dell'HoloLens (Figura 4.31). Premettendo che l'operazione richiede un'attenzione prolungata, se paragonata alla selezione di un bottone, da parte dell'utente era frequente che la mano uscisse dal gesture frame dando luogo a comportamenti non voluti del way point aumentando errori e frustrazione.

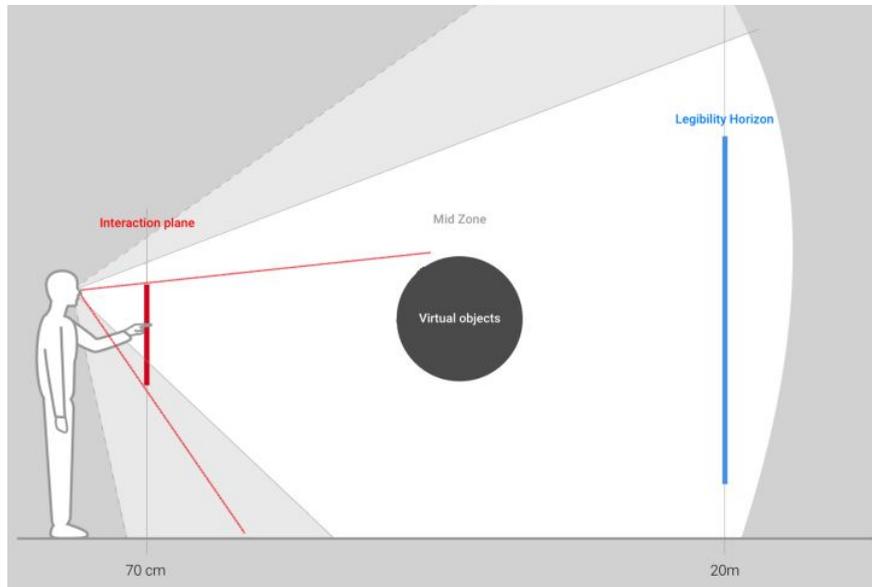


Figura 4.31 In rosso il gesture frame

La terza funzionalità permette di spostare il terreno. È stato deciso di mettere un controllo esterno al terreno stesso poiché con il terreno si interagisce in diversi modi per funzioni legate alla finalità dell'applicazione e non si voleva rischiare nel mezzo di una operazione di spostare l'ologramma accidentalmente. Data la natura irregolare del terreno, per aiutare l'utente a muovere il terreno, quando si abilita l'opzione viene visualizzato un piano virtuale come riferimento. Il piano è lo zero di riferimento per il terreno e permette di appoggiarlo su superfici reali (Figura).

Per quanto riguarda i menu contestuali hanno tutti la funzione di arricchire di informazioni e/o di opzioni il soggetto a cui sono connessi rivolgendosi sempre verso l'utente con la funzione billboard. Concentrandosi sul menu del field of view della camera del drone si vede come questo menu sia pensato per essere aperto per il tempo necessario ad impostare i parametri e poi chiuso non ostacolando più la vista dell'ambiente di lavoro. Il pannello prevede un riquadro in alto che permette di visualizzare cosa sta inquadrando la camera dell'UAV. Al di sotto di questo i vari parametri per i settaggi della camera. In questo, a differenza del discorso fatto per i way point, si è scelto di far sì che i parametri vengano impostati mediante il tracking della mano. Gli slider sono messi in posizione centrale tra la rappresentazione del FOV sul terreno e lo schermo della camera. In questo modo una volta selezionato il parametro d'interesse, l'operatore possa concentrarsi su una delle due

visualizzazioni mentre modifica il parametro vedendo comodamente l'effetto delle sue variazioni. In questo caso l'uscita della mano dal gesture frame causa un disagio minimo lasciando lo slider nell'ultima posizione registrata. L'operatore può dunque selezionarlo nuovamente e riprendere da dove aveva lasciato.

5 Valutazioni

Il sistema nel suo complesso consente una navigazione dinamica del territorio Piemontese e una prima istanza di pianificazione e valutazione di un volo per acquisizione di dati mediante un sensore. Nonostante possa produrre file GPX per la trasmissione dei way point a terze parti non fornisce pattern e/o analisi delle traiettorie di volo.

Tutti gli obiettivi prefissati sono stati raggiunti e da un punto di vista sperimentale sono state verificate con successo le possibilità di questo sistema. Avendo seguito uno sviluppo iterativo non tutte le funzioni sono state correlate tra loro ma si volevano provare più scenari possibili per valutare le possibilità della tecnologia in questione, si è comunque verificata la possibilità strutturale di implementarle in scenari di sviluppi futuri.

Nel complesso, gli aspetti che hanno riscosso un maggior interesse sono:

- la visualizzazione tridimensionale dei terreni e dei dati associati consentendo un netto miglioramento dell'intelligibilità dei dati ed eventuali situazioni da analizzare. Miglioramento riscontrato sia per gli addetti ai lavori che non, che si traduce in una riduzione dei tempi di interpretazione, ottimizzazione dei tempi decisionali e una miglior trasmissione di informazioni tra le due parti (soprattutto in un'ottica di multi-user);
- la pianificazione di voli, con particolare attenzione per quelli a corto raggio. Questi potrebbero trarre maggior benefici da una visualizzazione tridimensionale data la loro natura più soggetta a situazioni di volo miste, non solo planari rispetto al terreno;
- la possibilità di personalizzare l'ambiente di lavoro con elementi virtuali permettendo all'operatore di non essere più legato a dei monitor fisici per visualizzare le informazioni di interesse. Grazie alla Augmented Reality è possibile visualizzare quanti pannelli si vogliono posizionandoli nello spazio a proprio piacimento, mettendo le informazioni di rapida consultazione nelle zone più prossime al terreno e quelle a consultazione più sporadica in zone periferiche in base alle necessità.

Per quanto riguarda il device su cui si è scelto di sviluppare l'applicazione, ovvero l'Hololens, si è riscontrato un modico vantaggio nell'utilizzo della Mixed Reality in questo ambiente. L'unico vero utilizzo della MR consiste nella coerenza tra l'ambiente reale e quello virtuale dell'interfaccia grafica. Per il resto l'applicazione si collocherebbe più in un ambito di AR. Ciò nonostante, ad oggi, l'Hololens rimane il miglior device per la visualizzazione e l'interattività con ologrammi.

Si sono effettuati dei test interni all'azienda Altec con Hololens, su un campione di soggetti che non avevano esperienze pregresse con questo device. Nonostante l'impaccio iniziale con un nuovo metodo d'interazione ed un holografic frame percepito da tutti limitato, si è visto un alto interesse sulle potenzialità di questa tecnologia. Con particolare attenzione alla possibilità di visualizzare informazioni tridimensionali senza occludere la vista.

Da un punto di vista tecnico/tecnologico questo progetto si è trovato a doversi confrontare con le limitazioni e i bug dati da questa tecnologia, ancora molto giovane. Questo ha portato l'iter dello sviluppo a centrarsi più sul funzionamento delle singole funzioni piuttosto che sulla visione d'insieme, avendo ripercussioni anche sull'ottimizzazione.

Si andranno ora ad analizzare rapidamente i singoli argomenti:

GDAL

Per tutto il primo periodo si è cercato di integrare le librerie GDAL all'interno dell'applicazione su HoloLens. Anche avvalendosi dell'aiuto dei Microsoft Evangelist non si è riusciti a raggiungere questo obiettivo. Le librerie GDAL, non essendo native C#, mettono a disposizione un porting su C# mediante SWIG, strumento per connettere i programmi scritti in C e C++ a linguaggi di programmazione di alto livello. Questo strumento crea un wrapper alle dll originali utilizzando le *PInvoke* per connettere le librerie a C#. Questo metodo viene supportato dal framework .NET Framework ma non dal .NET Core su cui si appoggia l'HoloLens. Si è dunque optato per spostare l'utilizzo di queste librerie su lato server esterno. Nei mesi successivi pare che sia stato risolto il problema da parte di GDAL con la build di C# *"Support building gdal using the .NET Core 2.0 framework/SDK"* riporta il forum di supporto *"This commit adds solution and project files that support building the csharp binding dlls using the .NET Core 2.0 SDK"*. Al momento della risoluzione del problema l'applicazione era ormai strutturata in altro modo ed in valutazioni successive si sono trovati ulteriori vantaggi a mantenere le librerie su server esterno. Si nota comunque un certo tempo fisiologico di allineamento dei software di terze parti con le nuove linee guida di Microsoft e i UWP.

MixedRealityToolkit

Microsoft mette a disposizione un toolkit per lo sviluppo di applicazioni per la Mixed Reality. Contemporaneamente è stato messo a disposizione un toolkit nominato Mixed Reality Design Labs, basato a sua volta su MRTK ma con funzioni aggiuntive. Ad un certo punto dello sviluppo dell'applicazione HSR il MRTK ha integrato il MRLD e quest'ultimo è stato abbandonato. L'integrazione dei due sistemi ha causato dei cambiamenti nella struttura delle classi che gestiscono gli input rendendo necessaria una rivisitazione di una buona parte del codice sviluppato fino a quel momento. L'integrazione del nuovo MRTK era inoltre necessaria poiché, oltre ad avere le funzionalità aggiuntive del MRLD, risolveva numerosi bug delle funzioni implementate in questo progetto. Questi, relativamente, frequenti aggiornamenti costringono gli sviluppatori a dover decidere se rimanere con toolkit non aggiornati non usando le funzioni buggate o trovando un workaround, oppure aggiornarli sapendo però di dover riprendere in mano parte del codice.

Unity

Unity a sua volta sta implementando una serie di strumenti per lo sviluppo di applicazioni in MR e aggiungendo/adattando il proprio engine alle nuove esigenze. Venendo installato l'engine di Unity su HoloLens corrispondente alla versione usata per sviluppare il progetto il funzionamento di quest'ultimo è strettamente collegato alla versione di Unity usata. Anche in questo caso si è riscontrata un'alta variabilità tra le varie versioni, anche nei minor update (e.g. 2017.3.0.1 e la 2017.3.0.3). Portando di nuovo lo sviluppatore a dover scendere a

compromessi tra bug risolti e nuovi riscontrati. A tal proposito si vuole mettere in evidenza lo sviluppo delle classi per query http. In questo caso Unity metteva a disposizione la classe WWW. Dismessa in alcune versioni e poi reintegrata come wrapper di un'altra classe, UnityWebRequest. Entrambe le classi hanno sofferto di problemi per le applicazioni UWP in varie versioni di Unity rendendo impossibile capire quando il trasferimento dati fosse concluso. Sotto le versioni con problemi prese dal sito ufficiale di Unity^[13] per dare un'idea della saltuariet  del problema:

Unable to test with 2017.3.0a1 - 2017.3.0a7 due the errors
Reproduced with: 2017.3.0b1, 2017.3.0p2, 2018.1.0b2, 2018.2.0a1
Not reproduced with: 5.6.4p4, 2017.1.3f1, 2017.2.1p2
Fixed in: 2018.2.0a4
Backported: 2017.4.2f1, 2018.1.0b11

Nelle mentre Unity continuava a sviluppare la UnityWebRequest fornendo Handler diversi per tipologia di dati trasmessi (e.g. texture, multimedia, file) per demandare l'interpretazione dei dati ricevuti in base alla tipologia ad una thread diversa da quella principale. Fornendo un propriet  della classe "downloadProgress" che avrebbe assunto valore uguale a 0 finch  non si fosse completato il download, caso in cui sarebbe stata impostata ad 1. Qui sotto viene riportato il log di una query come esempio

```
[HSR_TextureWrapper.busyWait] wait iteration 0 (max: 120) unchanged: 0, downloadProgress: 0, isDone: False, bytes: 0^M
[HSR_TextureWrapper.busyWait] wait iteration 1 (max: 120) unchanged: 0, downloadProgress: 0, isDone: False, bytes: 0^M
[HSR_TextureWrapper.busyWait] wait iteration 2 (max: 120) unchanged: 0, downloadProgress: 0.5321451, isDone: False,
bytes: 92738^M
[HSR_TextureWrapper.busyWait] wait iteration 3 (max: 120) unchanged: 0, downloadProgress: 0.6826111, isDone: False,
bytes: 237920^M
.
.
.
[HSR_TextureWrapper.busyWait] wait iteration 21 (max: 120) unchanged: 0, downloadProgress: 0.5102293, isDone: False,
bytes: 2845398^M
[HSR_TextureWrapper.busyWait] wait iteration 22 (max: 120) unchanged: 0, downloadProgress: 0.5111564, isDone: False,
bytes: 2850568^M
[HSR_TextureWrapper.busyWait] wait iteration 23 (max: 120) unchanged: 1, downloadProgress: 0.5111564, isDone: False,
bytes: 2850568^M
[HSR_TextureWrapper.busyWait]          exiting loop: 25 (max: 120) unchanged: 2, downloadProgress: 0.5111564, isDone:
False, bytes: 2850568^M
```

Si pu  notare come la propriet  "isDone" rimanga sempre uguale a "false" e il "downloadProgress" assuma una serie di valori diversi da 1 o 0 e soprattutto non raggiunga mai l'1 e, in generale, non progredisca monotonamente, rendendo impossibile la sincronizzazione tra le funzioni basandosi sui metodi forniti dalla classe. Vi sono diversi casi come questo che costringono all'utilizzo di work-around con l'utilizzo di criteri euristici sostitutivi.

Performance

Dati i problemi sopra citati   stato citati   stato difficile fare effettuare dei test di performance sull'Hololens. Si   dunque cercato di ottimizzare l'applicazione basandosi su Unity. Dai primi test su Unity si   vista la necessit  di eliminare l'istanziamento di variabili nei

metodi “Update” dato il grosso impatto del garbage collector nelle performance.

Per migliorare i tempi di rendering e di attesa di risposta dalla rete si è optato per precaricare una unica texture per tutta la scena, a differenza dell’implementazione originale. Questa prevedeva di richiedere e caricare una texture per ogni singola Tile raggiungendo una risoluzione d’immagine superiore ma comportando uno stress non supportato dal device. Anche l’implementazione del LOD non è stata possibile poichè i terreni vengono caricati dinamicamente. In questo caso si sarebbero dovuti creare dinamicamente livelli di dettaglio diversi allungando esponenzialmente i tempi di attesi per l’avvio e la navigazione del terreno. Come si può vedere nel grafico qui sotto nell’ambiente di sviluppo di Unity (Figura 5.1)

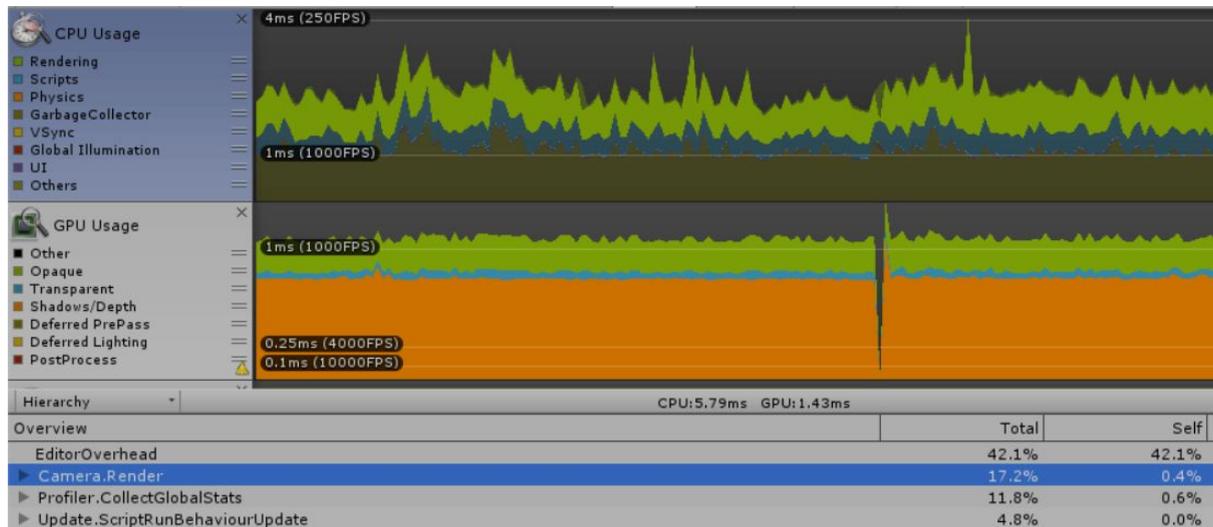


Figura 5.1 screenshot performance Unity

si raggiungono dei valori intorno ai 300FPS mentre su Hololens ci si aggira intorno ai 10FPS rendendo l’esperienza utente “scattosa” su operazioni medio-veloci e dando luogo a dei glitch mentre si muove lo sguardo.

Durante i test è stato rilevato un dato inatteso. Ovvero anche senza visualizzare il terreno, lasciando quindi solo l’interfaccia grafica principale sviluppata con il toolkit ufficiale, gli FPS non salgono sopra il valore 16 (Figura 5.2).



Figura 5.2 screenshot performance Hololens

Questi valori vanno comunque presi con le debite valutazioni. Come si può vedere nell'immagine precedente il processo che consuma più risorse è "EditorOverhead" riconosciuto ufficialmente da Unity come bug. Risulta quindi difficile riuscire ad identificare la causa precisa del crollo di prestazioni su Hololens.

In conclusione, la mia opinione su questa tecnologia è che non sia ancora pronta per l'utilizzo in ambito industriale per applicazioni articolate. Il potenziale è molto alto e sicuramente verrà raggiunto in poco tempo, molte aziende si stanno preparando ad uscire sul mercato con tecnologie simili e la stessa Microsoft dovrebbe far uscire la seconda versione dell'Hololens a breve. Bisogna tener conto che attualmente si lavora su un prodotto innovativo che ha già 3 anni ed un crescente interesse economico (che comporta la possibilità da parte delle aziende produttrici di investire maggiormente sulla tecnologia). Ad oggi è importante testarne capacità e limiti, in modo da poter valutare al meglio l'offerta del device che proporrà il mercato, ed essere in grado di progettare efficacemente le applicazioni sui nuovi dispositivi.

5.2 Sviluppi futuri

Il sistema permette svariate possibilità di sviluppi dal mission control, all'analisi di dati nel dopo missione.

Lo sviluppo più prossimo sarebbe la pianificazione di missioni. Caratterizzando in maniera specifica gli UAV, in modo avere informazioni di quota di lavoro e velocità di crociera, e aggiungendo i pattern di volo il sistema sarebbe in grado di fornire dei piani di volo più precisi. Per quanto riguarda i pattern di volo, in oltre, alcuni UAV necessitano solo del punto di interesse. In base allo scopo ed al punto di interesse sono in grado di calcolare le traiettorie da soli. A questo punto servirebbero solo all'operatore per avere un'idea migliore di ciò che farà il veivolo.

Fornendo un dataset di oggetti 3D georeferenziati sarebbe inoltre già possibile pianificare voli a corto raggio per acquisizione dati su edifici o strutture. Il sistema gestisce già la georeferenziazione, la creazione di mesh e acquisizione di dati da server. Quindi si potrebbero utilizzare dataset in remoto e con solo dati "grezzi" di scansioni con laser che forniscono una nuvola di punti per ricreare la struttura interessata.

Implementando un flusso di dati in realtime il sistema potrebbe svilupparsi in un mission control per controllare il volo e i dati ad esso correlati in tempo reale, confrontandolo anche con i percorsi attesi. Grazie alla visualizzazione tridimensionale permetterebbe una valutazione più rapida della situazione con annesso azione dell'operatore per correggere eventuali comportamenti non corretti o perfezionare il volo.

Sviluppando la possibilità di confrontare i dati di una stessa area/struttura acquisiti in momenti differenti si potrebbe sfruttare al meglio le potenzialità della visione tridimensionale per mettere in risalto i cambiamenti temporali. Permettendo così un'analisi più accurata ed una miglior percezione contestuale dell'elemento sotto analisi. Si potrebbe dunque valutare il

soggetto senza compromessi tra una visione bidimensionale, che può dare adito ad errori e/o percezioni fuorvianti, e la necessità di recarsi fisicamente sul posto, cosa a volte impossibile dato il difficile accesso di alcune zone.

Tutte queste implementazioni devono comunque tener conto dei limiti attuali dell'hardware e della potenza di calcolo ad esso connessa. Si suppone che nel relativamente breve periodo verranno immessi nel mercato la nuova generazione di dispositivi per la visione olografica. Se così non fosse si può sempre valutare di spostare la parte di calcolo su macchine esterne ed usare l'Hololens per la visualizzazione dei dati finali. A mio avviso il vantaggio della visione tridimensionale giustificherebbe la struttura del sistema eventualmente più complessa.

Bibliografia

- [1] Blumberg, S. (2018, May 30). 3D View of Amazon Forests Captures Effects of El Niño Drought. Retrieved from <https://www.nasa.gov/feature/goddard/2018/3d-view-of-amazon-forests-captures-effects-of-el-ni-o-drought>
- [2] Odom, J. (2017, February 20). Boeing Is Fighting Wildfires with the Microsoft HoloLens & Large Drones. Retrieved from <https://next.reality.news/news/boeing-is-fighting-wildfires-with-microsoft-hololens-large-drones-0175154/>
- [3] Alter, S. (1984). *Decision support systems: Current practice and continuing challenges*. Addison-Wesley.
- [4] Gorry, G. A., & S., S. M. (1971). *A framework for management information systems*. Massachusetts Institute of Technology.
- [5] Keenan, P. B. (n.d.). Spatial Decision Support Systems. *Decision-Making Support Systems*. doi:10.4018/9781591400455.ch003
- [6] Ivan E. Sutherland. (n.d.). Retrieved from <http://www.computerhistory.org/fellowawards/hall/ivan-e-sutherland/>
- [7] The Evolution of Augmented Reality –. (2017, October 19). Retrieved from <https://vydia.com/evolution-augmented-reality/>
- [8] Schmalstieg, D., & Höllerer, T. (2016). *Augmented reality: Principles and practice*. Addison-Wesley.
- [9] Commercial Drones from Intel. (n.d.). Retrieved from <https://www.intel.com/content/www/us/en/drones/drone-applications/commercial-drones.html>
- [10] Record over \$3B AR/VR investment in 2017 (\$1.5B in Q4). (n.d.). Retrieved from <https://www.digi-capital.com/news/2018/01/record-over-3b-ar-vr-investment-in-2017-1-5b-in-q4/#.W0YksdlzZEY>
- [11] Machine Intelligence Platform. (n.d.). Retrieved from <http://www.cbinsights.com/>
- [12] (n.d.). Retrieved from <http://www.lidar-uk.com/how-lidar-works/>
- [13] Technologies, U. (2017, December 25). Unity IssueTracker - [WSA] isNetworkError always return true when running an UWP app on x86 architecture. Retrieved from <https://issuetracker.unity3d.com/issues/wsa-isnetworkerror-always-return-true-when-running-an-uwp-app-on-86x-architecture>