POLITECNICO DI TORINO

Corso di Laurea
in Ingegneria Biomedica

Tesi di Laurea Magistrale

Sviluppo di un sistema mixed-reality (MR) come guida intraprocedurale per l'esecuzione di biopsie percutanee polmonari TC guidate

Relatori
prof. Gabriella Balestra
prof. Valentina Giannini
prof. Samanta Rosati

Candidato
Cristiana Di Egidio

Anno Accademico 2022-2023

A Carmelita, Vincenzo
e Sebastiano

Sommario

Negli ultimi anni la realtà aumentata, così come la realtà virtuale o la realtà mista, è tra le tecnologie maggiormente investigate in svariati ambiti, tra cui quello medico. Mentre la realtà aumentata consiste nella capacità di sovrapporre ambienti reali e virtuali in tempo reale, la realtà virtuale fornisce all'utente un'esperienza esclusivamente immersiva. La realtà mista, invece, si pone a metà strada tra un mondo interamente digitale e il mondo reale e consente l'interazione tra i due: si può posizionare un oggetto virtuale nell'ambiente reale come se esso fosse fisicamente presente, o viceversa si può essere presenti nel mondo reale in modo digitale per collaborare con terze parti in momenti e luoghi diversi.

In particolare, l'introduzione di tali tecnologie comporta dei vantaggi come l'ottimizzazione di tempistiche e procedure, può essere d'aiuto nella formazione di nuovo personale medico, può facilitare la visualizzazione di dati relativi al paziente e fornire supporto durante interventi chirurgici. La maggiore diffusione di applicazioni di realtà aumentata e virtuale è dovuta ai recenti progressi tecnologici, che hanno comportato la disponibilità di componenti hardware e software con elevata potenza di calcolo, la miniaturizzazione di componenti elettroniche e, di conseguenza, anche un ampio sviluppo di dispositivi indossabili.

In questo lavoro di tesi è stata sviluppata un'applicazione di realtà aumentata distribuita sul visore Microsoft Hololens 2 e finalizzata a guidare la procedura medica di biopsia transtoracica. Infatti, l'idea di base è quella di ottenere sul visore il volume 3D del torace del paziente, ricostruito a partire dalle immagini TC. In questo modo sarebbe possibile superare il limite principale della procedura standard, cioè la necessità del medico di effettuare un continuo passaggio tra la visualizzazione delle immagini TC e il corpo del paziente. Inoltre, durante l'esecuzione dell'intervento, è necessario effettuare più scansioni proprio per verificare che la traiettoria dell'ago sia corretta e che la lesione target sia stata centrata con successo, sottoponendo il paziente a una maggior dose di radiazioni. Invece, seguendo la procedura sperimentale proposta, il medico dovrebbe effettuare la biopsia transtoracica visualizzando costantemente sul visore il torace del paziente ricostruito tridimensionalmente e sovrapposto a quello reale.

In un'applicazione del genere l'aspetto più critico, che determina la sicurezza

della procedura medica, è costituito dalla necessità di avere una perfetta corrispondenza spaziale tra il volume visualizzato sul visore e il volume reale dato dal torace del paziente. In questo lavoro si è cercato di ottenere un matching spaziale tra i due mediante l'ancoraggio del volume con QR code. Dunque, è stato utilizzato un fantoccio di un emitorace stampato 3D ricoperto di gommapiuma con all'interno delle palline che simulano delle ipotetiche lesioni del polmone, in modo da avere un modello anatomico semplificato rispetto alla realtà. Tale volume è stato ricostruito a partire dalle immagini TC sul software 3D Slicer, successivamente è stato importato sul software Unity 3D in cui è stato sviluppato il resto dell'applicazione. Infine, il progetto è stato compilato e distribuito sul dispositivo Microsoft Hololens 2 mediante Visual Studio 2019.

In particolare, il primo capitolo fornisce una panoramica generale sulla realtà aumentata, e illustra l'applicazione di tale tecnologia in diversi settori, concentrandosi principalmente su quello medico. Verrano riportati infatti numerosi esempi di applicazioni AR ad uso medicale sviluppate con finalità diverse.

Nel secondo capitolo vengono trattate le tecnologie necessarie per creare un'applicazione basata sulla realtà aumentata in ambito medico; quindi, vengono introdotti i software che sono stati successivamente usati nello sviluppo del progetto.

Il terzo capitolo invece si concentra sulla trattazione della parte sperimentale del progetto di tesi, viene spiegata la procedura svolta a partire dall'ottenimento delle immagini TC del fantoccio fino alla distribuzione dell'applicazione sul visore.

Il quarto capitolo analizza i risultati ottenuti testando l'applicazione nel reparto di radiologia dell'ospedale Molinette di Torino, tratta i rischi legati a tale applicazione e i suoi possibili sviluppi futuri.

Indice

Elenco delle figure 7

1 Introduzione 9
- 1.1 Introduzione alla realtà aumentata . . . 9
- 1.2 Applicazioni AR . . . 10
- 1.3 Utilizzo dell'AR in campo medico . . . 12
  - 1.3.1 Componenti di un sistema AR medicale . . . 12
  - 1.3.2 Applicazioni di VR e AR in istruzione e formazione . . . 13
  - 1.3.3 Applicazioni AR in chirurgia e radiologia . . . 14
  - 1.3.4 Vantaggi e limiti di un sistema AR ad uso medicale . . . 18

2 Tecnologie per lo sviluppo di un software AR ad uso medicale 19
- 2.1 Rendering volumetrico mediante Slicer 3D . . . 19
- 2.2 Unity 3D . . . 22
- 2.3 Mixed Reality Toolkit (MRTK) . . . 24
- 2.4 Microsoft Hololens 2 . . . 25
  - 2.4.1 Caratteristiche principali di Hololens 2 . . . 25

3 Descrizione del progetto 29
- 3.1 Procedura standard . . . 29
- 3.2 Procedura sperimentale . . . 30
  - 3.2.1 Ottenimento del volume . . . 30
  - 3.2.2 Ancoraggio del modello . . . 33
  - 3.2.3 Implementazione del progetto su Unity 3D . . . 34

4 Risultati ottenuti 43
- 4.1 Analisi del rischio . . . 45
- 4.2 Sviluppi futuri . . . 46

5 Conclusioni 49

Elenco delle figure

1.1 Rappresentazione del virtuality-continuum. 9
1.2 Applicazioni AR: (a) servizi, (b) principali aree di applicazione. . . 11
1.3 (a) Rovine del tenpio. (b) Ricostruzione tempio mediante l'AR. . . 12
1.4 (a) Modalità AR, (b) modalità VR. 13
1.5 Simulazione AR. 14
1.6 Modello di un organo 3D sovrapposto a un'immagine laparoscopica. 16
1.7 L'interfaccia di un software di pianificazione del trattamento preoperatorio con visualizzazione 3D. 17
2.1 Strutture vascolari polmonari evidenziate dalla MIP. 20
2.2 Esempi di scale di grigio usate comunemente in 3D Slicer. 22
2.3 Unity Hub con diverse versioni del software installate. 23
2.4 Interfaccia grafica di Unity con i suoi vari elementi evidenziati. . . . 23
2.5 Microsoft Hololens 2. 26
3.1 Fantoccio stampato 3D ricoperto di gommapiuma. 30
3.2 Immagini formato DICOM caricate su 3D Slicer. 31
3.3 3D Slicer Editor: selezione del tipo di volume. 32
3.4 Volume ottenuto in formato STL. 32
3.5 Modello 3D ottenuto su Unity 33
3.6 QR code utilizzato. 34
3.7 Parametri impostati su Project Settings. 35
3.8 QRCode Panel visto in Hierarchy e Scene. 36
3.9 QR Code Manager visto dall'Inspector. 37
3.10 Parametri impostati in Built Settings. 38
3.11 Parametri imposti per compilazione e distribuzione su Visual Studio 2019. 39
3.12 Pannello dopo aver scannerizzato QR code non desiderato. 40
3.13 Corrispondenza tra modello reale e virtuale. 40
3.14 Pannello con le informazioni relative al QR code voluto. 41
4.1 Simulazione procedura sperimentale. 43
4.2 Risultati ottenuti: confronto tra i due metodi. 44
4.3 Possibile configurazione con 3 QR code. 47

Capitolo 1

Introduzione

1.1 Introduzione alla realtà aumentata

La realtà aumentata (AR) consiste nella capacità di sovrapporre elementi virtuali con la realtà fisica nello spazio tridimensionale, mediante un'interazione istantanea tra i due. Essa è una variante degli ambienti virtuali (VE), più comunemente identificati con il termine realtà virtuale, ed è a sua volta una sottocategoria del più ampio gruppo identificato come realtà mista. Infatti, nel caso della realtà virtuale, l'utente è completamente immerso in un ambiente sintetico, mentre nel caso dell'AR oggetti sintetici coesistono con oggetti reali, e questi ultimi sono preponderanti nell'ambiente.

Figura 1.1. Rappresentazione del virtuality-continuum.

Molti autori hanno definito l'AR come un sistema che necessita dell'utilizzo di un dispositivo a montaggio frontale (HMD), ma in maniera più generale e meno limitante dal punto di vista tecnologico, la realtà aumentata può essere definita come un sistema con le seguenti caratteristiche[1]:

1. combina mondo reale e virtuale;

2. è un sistema interattivo in tempo reale, infatti gli oggetti reali e virtuali devono essere fusi tra loro in modo da dare all'utente l'idea di trovarsi in un unico ambiente;

3. l'interazione tra mondo reale e virtuale e il funzionamento del sistema avvengono all'interno dello spazio tridimensionale.

Tale definizione ammette comunque l'utilizzo di dispositivi mobili o interfacce basate su monitor. Attualmente l'avvento dell'AR e il suo graduale passaggio dagli ambienti di laboratorio al campo applicativo è dovuto principalmente allo sviluppo tecnologico crescente negli ultimi anni, in particolare è dato principalmente dall'aumento della potenza di calcolo delle parti hardware e software del computer, dalla maggiore diffusione di dispositivi mobili e indossabili, dalla disponibilità di dati geo-referenziati e dalla facile accessibilità a reti mobili ad alta velocità.[2]

Gli elementi principali di tale tecnologia sono:

- **tecnologia di visualizzazione intelligente**: vi sono tre categoria di display intelligenti. I primi sono gli HMDs che permettono di fondere in tempo reale immagini reali e sintetiche, e loro volta sono suddivisibili principalmente in display ottici prospettici e video prospettici. Poi vi sono i display dei dispositivi portatili, ed infine vi sono i display dei Pc desktop che riescono a combinare insieme il mondo reale registrato dalla telecamera del PC con un mondo virtuale generato dal PC stesso;

- **tecnologia di registrazione 3D**: permette l'effettiva sovrapposizione di immagini reali e virtuali. La registrazione 3D può essere basata sulla computer vision, su una rete wireless, su un tracker hardware o su un mix delle tecnologie sopracitate;

- **tecnologia di interazione intelligente**: grazie a tale tecnologia è possibile avere interazione tra persone e oggetti visrtuali su una scena reale. Infatti l'utente può gestire la propria esperienza dando istruzioni agli oggetti virtuali e a sua volta riceve un feedback da essi. Esempi di interazioni intelligente sono interazioni basate sulla posizione, interazioni con dispositivi hardware o basate su informazioni varie.[3]

1.2 Applicazioni AR

Molti servizi ingegneristici, per essere eseguiti al meglio, richiedono l'utilizzo di procedure standardizzate, necessitano di molte informazioni che mediante l'AR sono più facilmente reperibili e devono essere svolti sul campo. A tal proposito, l'AR rappresenta un buon alleato a cui va affiancato un elevato numero di sensori,

che permettono di reperire maggiori informazioni sull'ambiente in cui si opera. Il rilevamento dei dati deve essere strattamente legato all'AR, e qualunque cambiamento deve essere recepito in tempo reale. Il metodo più utilizzato è la tecnologia video mixting che permette all'utente di svolgere un'esperienza più immersiva grazie ai sistemi HDM. L'AR è una tecnologia che si presta a vari campi applicativi, alcuni già collaudati da anni, altri in via di sviluppo più recente.[4]

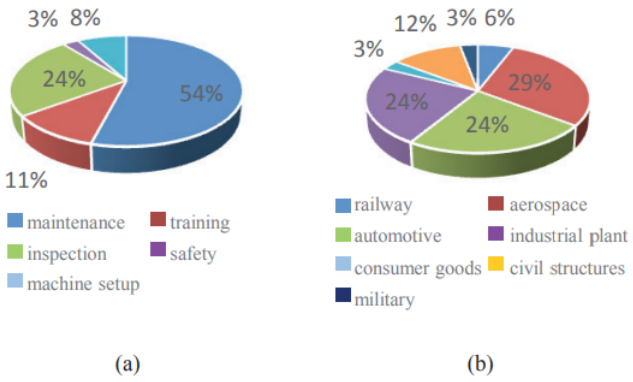

Figura 1.2. Applicazioni AR: (a) servizi, (b) principali aree di applicazione.

Ad esempio, i servizi di manutenzione e riparazione sono principalmente rivolti ai campi dell'automotive e dell'ingegneria aereospaziale, ma sono diffusi anche nel campo dell'ingegneria civile per il mantenimento di infrastrutture come le ferrovie. Anche nel settore militare è stato investigato l'uso dell'AR, che risulta utile ad esempio nella simulazione di scenari di guerra o per migliorare le condizioni di visualizzazione di un pilota d'aerei militari. In ambito archeologico invece l'AR permette di migliorare l'esperienza del turista e massimizzarne la conoscenza: un esempio di ciò è costituito dal progetto Archeoguide, nel quale, mediante un sistema mobile, progettato in modo sofisticato e in grado di fornire una guida elettronica al turista, viene effettuata una ricostruzione 3D del sito archeologico di Olimpia.[5]

Altri esempi di aree applicative, in cui il supporto dell'AR è crescente, sono: l'industria, in cui possono essere migliorati i processi produttivi e la qualità dei prodotti; l'istruzione, in cui grazie all'AR si ha la possibilità di visualizzare concetti astratti e migliorare quindi l'apprendimento; la robotica e la medicina. [6] In particolare l'utilizzo dell'AR in campo medico verrà discusso più nel dettaglio nel seguente paragrafo.

Figura 1.3. (a) Rovine del tenpio. (b) Ricostruzione tempio mediante l'AR.

1.3 Utilizzo dell'AR in campo medico

La realtà mista, in particolare l'AR e la VR, è in fase di espansione da diversi anni anche in campo medico, ed è un argomento su cui si sono focalizzate molte ricerche sperimentali negli ultimi anni. In medicina vi è ancor più la necessità di dispositivi mobili leggeri, piccoli, facilmente maneggevoli e confortevoli per l'utilizzatore ma anche convenienti dal punto di vista economico. A tal proposito, un ruolo chiave per ottenere la diffusione della realtà mista, è stato svolto dai progressi tecnologici che hanno permesso anche una miniaturizzazione dei dispositivi, dalla disponibilità di software open source e dal vasto uso degli smartphone anche da parte del personale medico. In particolare, la VR fornisce un ambiente completamente immersivo, mentre l'AR sovrappone immagini reali con immagini olografiche, quindi, in ambito medico, spesso fornisce all'utente la percezione di vedere attraverso il corpo del paziente. Le differenze tra le due hanno fatto sì che tali tecnologie venissero rispettivamente usate per scopi diversi in ambito clinico; successivamente si vedranno le principali applicazioni dell'AR in campo medico. Nella seguente figura è possibile osservare l'esempio di un sistema che può essere usato sia in modalità AR che VR.[7]

1.3.1 Componenti di un sistema AR medicale

Nella progettazione di un sistema AR ad uso medico bisogna necessariamente eseguire dei passaggi, come la registrazione di informazioni quali i dati del paziente, l'importazione di dataset medici, e bisogna effettuare il tracciamento della telecamera utilizzata dal sistema AR e degli strumenti chirurgici utilizzati. Più nel dettaglio quindi, gli aspetti fondamentali di un sistema AR medicale sono:

- **Tracciamento della telecamera e degli strumenti**: è necessario infatti conoscere in tempo reale la posizione spaziale della telecamera e, in campo chirurgico, dev'essere nota anche la posizione della strumentazione usata dal

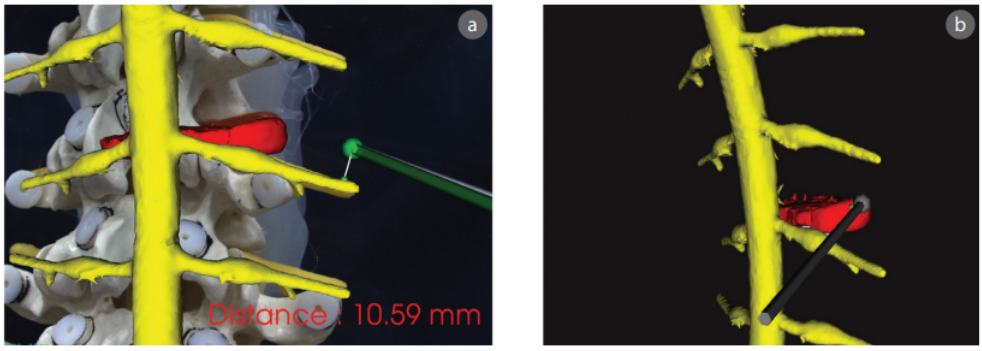

Figura 1.4. (a) Modalità AR, (b) modalità VR.

medico, bisogna inoltre definire le caratteristiche della telecamera utilizzata e trasformare gli oggetti virtuali rispetto alle coordinate del mondo reale;

- **Registrazione del paziente**: il paziente dev'essere registrato rispetto alle coordinate del mondo reale;

- **Dati di pianificazione preoperatoria**: i dati a partire dai quali vengono prese decisioni mediche provengono principalmente da immagini di tomografia computerizzata (TC) o di risonanza magnetica (RM) e vengono salvati nel formato standard DICOM. Per migliorare l'accuratezza di tali dati, si possono prendere dei punti superficiali a partire dalle immagini 3D, e a partire da questi si possono costituire traiettorie o modelli tridimensionali. Esistono software open source che permettono di effettuare la conversione delle immagini mediche dal formato DICOM in modelli 3D.[8] [9]

1.3.2 Applicazioni di VR e AR in istruzione e formazione

La VR e l'AR risultano particolarmente efficienti per quanto riguarda l'istruzione e la formazione di personale medico. Esse permettono infatti di superare alcune problematiche ricorrenti con i metodi educativi tradizionali, come ad esempio la difficoltà di visualizzare parti anatomiche particolarmente complesse o la penuria di cadaveri disponibili per lo studio dell'anatomia.

Inoltre, mediante tali tecnologie, è possibile fornire un'istruzione sempre più standardizzata ed egualitaria anche per studenti provenienti da ambienti più disparati. In aggiunta, è stato dimostrato che l'apprendimento mediante l'immersione in un ambiente 3D permette di ottenere una maggior partecipazione dell'utente, quindi si ottiene un'acquisizione di informazioni più veloce, e vi sono anche una

maggior collaborazione e interattività tra studenti. In particolare, sono molti i centri medici univeritari che hanno usato tali tecnologie negli Stati Uniti. Ad esempio, la VR è stata utilizzata per miglioare l'apprendimento dei vari step di procedure di radiologia interventistica, procedure endoscopiche, è stata usata anche per sviluppare terapie per pazienti con dolori cronici e per simulazioni di interventi effettuati in laparoscopia. L'AR a sua volta è stata utilizzata ad esempio per l'insegnamento dell'anatomia, per la visualizzazione in tempo reale del corpo del paziente durante procedure invasive, per la creazione di simulazioni in cui gli ologrammi vengono sovrapposti a dei manichini e per altre svariate applicazioni cliniche.[10].

Un esempio di simulazione è stato realizzato dalla Scuola di Infermierisica dell'Università del Michigan nel 2017: qui mediante l'utilizzo di un sistema AR e del dispositivo HoloLens (Microsoft 2020) è stata realizzata una simulazione dei primi sintomi di ictus,i quali, se riconosciuti tempestivamente, riducono il rischio di riportare gravi danni per il paziente. La valutazione dei primi sintomi di ictus comprende un'analisi della simmetria facciale, della capacità di linguaggio e della perdita di forza nelle braccia, e nel caso di tale simulazione è stato riportato che il 91,8% degli studenti ha riconosociuto con successo i sintomi.[11]

Figura 1.5. Simulazione AR.

1.3.3 Applicazioni AR in chirurgia e radiologia

Grazie agli sviluppi tecnologici degli ultimi anni, alla crescente disponibilità di immagini mediche di varie tipologie e l'accesso in tempo reale a tali risorse, si è

visto che in medicina il processo di diagnosi e trattamento sono diventati più veloci. Inoltre in chirugia è fondamentale poter avere a disposizione immagini mediche sia bidimensionali che 3D, e a tal proposito, l'AR cerca di realizzare un connubio tra la chirurgia ed immagini generate al computer.

A livello pratico, il primo passo fondamentale da compiere è l'allineamento tra le immagini generate al computer e la realtà, e ciò può essere svolto con diversi metodi di tracciamento, molti dei quali sono disponibili a livello commerciale. I più diffusi sono i sistemi di tracciamento elettromagnetici o a infrarossi, con tali sistemi vengono tracciati dei marcatori posti in dei siti specifici che rimangono fissi anche durante l'intervento chirurgico. Quindi, nelle tempistiche generali va considerato anche il tempo di calibrazione del sistema AR e della ricostruzione 3D delle immagini in modo che si adattino al display. Sebbene per quanto riguardi la visualizzazione di parti anatomiche più complesse l'AR sia di grande aiuto, a livello chirurgico, data la necessità di un sistema di tracciamento, risulta più complesso intervenire su organi mobili come ad esempio quelli addominali. Infatti, molti lavori di ricerca di sono concentrati sullo sviluppo di sistemi AR finalizzati a zone anatomiche come il cervello, la scatola cranica, l'apparato scheletrico e il pancreas. [12]

Inoltre vi sono anche lavori di ricerca in cui si indaga la possibilità di integrare sistemi di chirurgia robotica con la realtà aumentata, ad esempio in interventi eseguiti in laparoscopia o in esami endoscopici. In chirurgia robotica si ha bisogno di molte informazioni sul campo operatorio, in quanto il medico non vede direttamente l'interno del corpo del paziente, ma lo vede attraverso una camera o un sistema ottico che si trovano sul campo operatorio, per cui si ha anche una visuale più limitata rispetto ad altri tipi di interventi chirurgici. Pertanto può essere conveniente integrare un sistema a realtà aumentata con un robot chirurgico. Nell'articolo ivi citato si è tentato di combinare il classico robot Da Vinci con l'AR. Il Da Vinci è un sistema usato principalmente in chirurgia generale, urologica, vascolare, ginecologica o in classici esami endoscopici come la gastroscopia e la colonscopia. Tale sistema, integrato con l'AR e usato principalmente per interventi laparoscopici, è quindi costituito di dua parti: un sensore ottico che misura la posizione del laparoscopio, e una workstation grafica che cattura le immagini durante la laparoscopia e le sovrappone a un modello dell'organo 3D. In figura è possibile vedere la sovrapposizione dell'immagine al modello dell'organo 3D in una colecistectomia. [13]

Negli ultimi anni, oltre alle tradizionali tecniche chirurgiche, c'è stata anche un'ampia diffusione di terapie guidate dalle immagini (image-guided therapy) e di interventi di chirurgia mininvasiva (procedure endovascolari). Generalmente il medico durante l'esecuzione di una procedura deve costantemente passare dal vedere l'immagine al trovare il punto corrispondente nel corpo del paziente, per cui l'intervento è difficoltoso ed è influenzato dall'abilità dell'operatore, per cui il

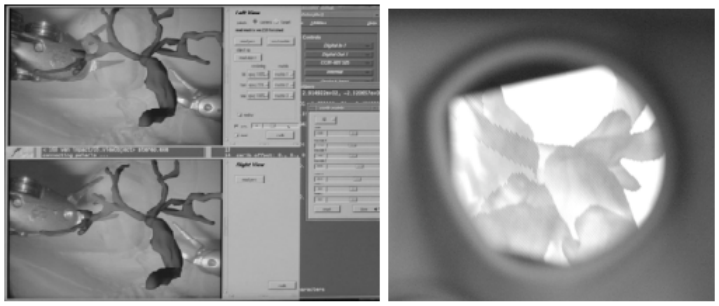

Figura 1.6. Modello di un organo 3D sovrapposto a un'immagine laparoscopica.

paziente è soggetto a dei rischi. Tali criticità possono essere superate grazie all'utilizzo di un sistema AR o VR, in quanto il medico non dovrebbe più effettuare un continuo cambio di focalizzazione dello sguardo, poichè riuscirebbe a visualizzare tutto contemporaneamente.

Ad esempio nel caso di procedure endovascolari, prima dell'intervento, il radiologo interventista può simulare gli angoli e le posizioni fluoroscopiche ideali per evidenziare il decorso dei vasi e i punti di diramazione, mentre durante l'intervento grazie all'aiuto di un sistema AR la dose di radiazioni può essere notevolmente ridotta. Infatti è stato dimostrato che usando il tracciamento elettromagnetico e un sistema AR è possibile sovrapporre un modello vascolare 3D su un paziente fantasma, tracciare e navigare virtualmente un catetere endovascolare attraverso il modello vascolare. In questo modo si elimina la necessità di radiazioni per la guida endovascolare in tempo reale.[14]

Sistemi AR sono risultati vantaggiosi e recentemente sono stati implementati anche per interventi percutanei con ago. Infatti si possono creare delle traiettorie virtuali dell'ago in modo da ottenere un posizionamento all'interno del corpo del paziente il più preciso possibile. Inoltre, è stato dimostrato che l'AR riduce i tempi di procedura, il numero di immagini acquisite e la dose di radiazioni durante diverse procedure come interventi ossei percutanei simulati, biopsie epatiche guidate da TC, ablazione percutanea del fegato. Dunque sistemi AR possono aiutare a pianificare il posizionamento e la traiettoria di una sonda, inoltre permenttono una migliore visualizzazione e localizzazione dei margini tumorali.

Ad esempio sono stati dimostrati tassi di successo di ablazione più elevati in simulazione di ablazione epatica a microonde dopo la pianificazione su modelli 3D, ed è stato riscontrato che la visualizzazione di traiettorie pianificate precendentemente può dare maggiore fiducia al radiologo interventista per avvicinarsi e trattare bersagli in posizioni difficili che in precedenza erano sfavorevoli, come le lesioni della cupola epatica.[15] [16]

Le immagini mediche, oltre che per l'approccio chirurgico, sono fondamentali

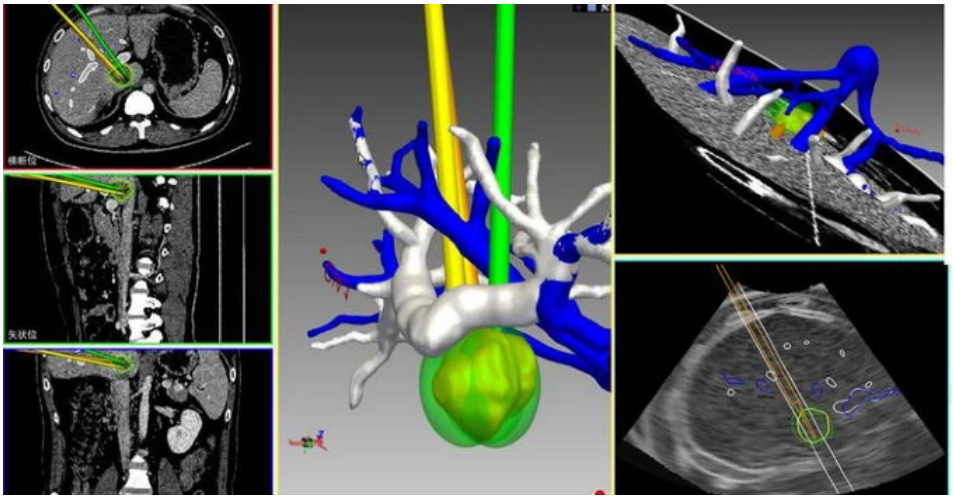

Figura 1.7. L'interfaccia di un software di pianificazione del trattamento preoperatorio con visualizzazione 3D.

anche dal punto di vista diagnostico. La loro precisione e la loro corretta interpretazione da parte del medico giocano un ruolo chiave nella tempestività di alcune diagnosi, come quelle del cancro, e di conseguenza anche nel trattamento del paziente. Infatti avere una diagnosi quanto più precoce possibile e quanto più corretta è fondamentale per il trattamento del paziente, e comporta dei benefici anche dal punto dei vista dei costi sostenuti per la sua cura. Convenzionalmente le immagini mediche utilizzate per visualizzare i volumi anatomici sono la tomografia computerizzata (TC), la risonanza magnetica (MRI) e la tomografia a emissione di positroni (PET). Con le attuali tecnologie è possibile effettuare un'acquisizione di grandi volumi nel giro di pochi secondi, e la ricostruzione volumetrica avviene "fetta per fetta" (slice). L'approccio tradizionale attualmente presenta delle criticità, come la grande quantità di informazioni da gestire: data l'elevata risoluzione spaziale raggiungibile, per ogni singola fetta di immagine ottenuta, il radiologo deve gestire un'elevata mole di dati (ad esempio una TC di una singola fetta di torace comporta la formazione di un dataset di 131 milioni di pixel), e inoltre può risultare abbastanza difficoltoso ricostruire volumi complessi a partire da slice 2D. Un altro aspetto da considerare è la difficoltà di identificare piccole lesioni come piccoli tumori in fase di formazione iniziale: questo fattore è di vitale importanza, in quanto la diagnosi precoce di un tumore comporta una più alta probabilità di sopravvivenza da parte del paziente. Pertanto per migliorare tali criticità possono essere d'aiuto software D3D in cui viene effettuato un rendering volumetrico a partire dalle immagini 2D, le quali vengono convertite in modo da ottenere una vista tridimensionale. Il volume 3D ottenuto è inoltre segmentato in modo da poter

distiguere meglio i diversi tipi di tessuto, e mediante il software vengono calcolate anche una vista destra e sinistra delle immagini in modo analogo a quanto fa il cervello con la vista umana. Tali tecnologie non sono ancora state approvate per uso medico ma risulterebbero efficaci soprattutto in oncologia, e le ricerche sperimentali sull'uso di AR e VR in radiologia sono via via crescenti, in quanto aprirebbero nuovi orizzonti per quanto riguarda l'efficacia di una diagnosi.[17] [18]

1.3.4 Vantaggi e limiti di un sistema AR ad uso medicale

Si è visto come negli ultimi anni sia di grande interesse l'applicazione di AR e VR in svariati ambiti tra cui quello clinico. In particolare tali tecnologie rappresentano una potenziale risorsa rispetto ai metodi tradizionali, in quanto, grazie ad esse, è possibile integrare anche feedback sonori e tattili oltre quello visivo, e inoltre il medico può facilmente interagire con il sistema mediante delle gesture. Tuttavia è fondamentale che l'utente provi un elevato comfort utilizzando ad esempio dei dispositivi HMD, e ciò non sempre si verifica in quanto è stato riportato che durante l'uso di applicazioni AR/VR si possano presentare nausea, mal di testa , sensazione di vertigini soprattutto se vi è un uso prolungato.

Uno dei limiti principali dei display AR è il campo visivo (FOV). Naturalmente, il FOV binoculare degli occhi umani è di circa 200° sul piano orizzontale e di 135° sul piano verticale. Tutti gli OST-HMD disponibili in commercio hanno FOV orizzontali o verticali inferiori a 90°, con la maggior parte di essi si colloca tra i 30° e i 40°.

Inoltre tali applicazioni, per essere sicure ed efficienti almeno quanto i metodi tradizionali, necessitano di un'elevata potenza di calcolo, ingenti risorse e costi non sempre sostenibili dalle aziende sanitarie. Vi sono già alcune applicazioni cliniche approvate dall'FDA e altre in fase sperimentale, ma alcune difficoltà riguardanti il loro effettivo utilizzo riguardano il trattamento dei dati del paziente. Infatti, ci sono dei rischi dal punto di vista legale per quanto riguarda il possibile furto di identità del paziente e l'utilizzo improprio dei suoi dati; è difficile per un sistema soddisfare gli standard richiesti da un'azienda sanitaria e usare dati crittografati non costituisce una totale garanzia. Inoltre, secondo alcune leggi non sarebbe possibile diffondere a terzi i dati confidenziali che vengono scambiati tra medico e paziente. Dunque, nonostante i numerosi vantaggi e i nuovi orizzonti esplorabili grazie all'introduzione di sistemi AR/VR in ambito clinico, le criticità sopra elencate andrebbero superate e tenute a mente nello sviluppo di tali applicazioni.[7] [19]

Capitolo 2

Tecnologie per lo sviluppo di un software AR ad uso medicale

2.1 Rendering volumetrico mediante Slicer 3D

I progressi tecnologici riguardanti le tecniche di imaging come la risonanza magnetica (MRI) e la tomografia computerizzata (TC) hanno avuto come conseguenza il dover gestire una grande mole di dati, e inoltre l'interpretazione di tali immagini è anche legata all'esperienza del medico. Il rendering volumetrico, cioè la ricostruzione del volume tridimensionale a partire dalle immagini, può aiutare a semplificare lo studio radiologico. Il volume ricostruito è formato da una serie di fette (slice) equidistanti e parallele. Nel caso di volumi ottenuti a partire da immagini tomografiche e di risonanza magnetica, ogni singola slice viene salvata in un file formato DICOM. Il volume dei dati ottenuto a partire dalle immagini è formato da una serie di voxel, cioè dei volumi elementari la cui base è data dalle dimensioni dei pixel della singola immagine, mentre l'altezza è pari alla distanza presente tra una slice e l'altra. Idealmente bisognerebbe effettuare la ricostruzione volumetrica a partire da voxel cubici, e a tal proposito vi sono algortmi che prima di effettuare la ricostruzione verificano le caratteristiche del volume, altrimenti non effettuano la ricostruzione in quanto si avrebbe una degradazione dei dati di partenza.

Esistono diversi algoritmi che si occupano di effettuare la ricostruzione volumetrica tridimensionale. Tali algoritmi possono essere suddivisi in modo più generale

in due macrocategorie: metodi di rendering diretti e metodi indiretti (**direct rendering** e **indirect rendering**). Nel primo caso, sono utilizzate tutte le informazioni di partenza delle immagini, non vi sono passaggi intermedi come ad esempio la segmentazione, dunque vi è il diretto passaggio alla rappresentazione tridimensionale. Tali tecniche si basano sull'intensità dei voxel, quindi un possibile contro è il fatto che si potrebbe ottenere una visualizzazione di strutture non appartenenti a quella voluta poichè possono esistere strutture con stessi valori di intensità (ogni voxel ha associato un livello di grigio), inoltre, conservando tutta l'informazione di partenza, ci si trova a gestire una grande quantità di dati. Tuttavia, il vantaggio di tali metodi sta proprio nella conservazione di tutte le informazioni iniziali che quindi vengono mantenute anche nella ricostruzione tridimensionale.

Una variante del volume rendering diretto è la tecnica **MIP** (Maximum Intensity Projection - massima intensità di proiezione). Tale metodo valuta valuta ogni voxel lungo una linea che parte dall'occhio dell'osservatore e attraversa il volume di dati, seleziona il valore massimo del voxel, che viene poi utilizzato come valore visualizzato. L'utilità clinica delle immagini MIP è stata ampiamente valutata, e la tecnica MIP si è dimostrata particolarmente utile nella sua applicazione originale per la creazione di immagini angiografiche ottenute a partire da una serie di TAC e RM.[20] [21]

Figura 2.1. Strutture vascolari polmonari evidenziate dalla MIP.

Nei metodi indiretti invece i dati volumetrici iniziali subiscono un passaggio intermedio, ovvero la conversione in primitive geometriche e un successivo rendering volumetrico ottenuto con tecniche poligonali. I vantaggi di tali tecniche sono la

velocità di realizzazione del rendering e la flessibilità, mentre il principale svantaggio consiste nella perdita di dati che si verifica nei passaggi intermedi, per cui, alla fine, nel peggiore dei casi i volumi ottenuti potrebbero anche portare a commettere degli errori dal punto di vista diagnostico.

3D Slicer è una piattaforma open source con accesso libero che consente di processare e analizzare immagini mediche, inizialmente nata per l'analisi di terapie guidate in campo neurochirurgico, e nel corso del tempo è stata ampliata con maggiori funzionalità. Tale software è ampiamente utilizzato per analisi cliniche e precliniche, ma anche in ambiti diversi da quello medico, e si differenzia da altri software commerciali e workstation che svolgono le stesse attività grazie alle sue caratteristiche principali. Esso infatti è dotato di un'ampia funzionalità, di license software illimitate, e di indipendenza tra le varie piattaforme. Inoltre, 3D Slicer è un software altamente fruibile in quanto si può installare liberamente in sistemi operativi Windows, Mac OS X e Linux e può essere anche ampliato con l'aggiunta di pacchetti specifici, mentre altri dispositivi commerciali di tipo slicer richiedono a volte particolari hardware. Un'ulteriore caratteristica da considerare di 3D Slicer è il fatto che il software non è approvato dall'FDA, per cui è l'utente a doversi assumere eventuali responsabilità di tipo etico ed a stabilirne la sicurezza. Il software supporta diverse tipoligie di immagini: PET, TC, MRI e ultrasuoni e tali immagini devono essere salvate nel formato standard DICOM per poter essere importate sulla piattaforma. Una volta importate le immagini, per effettuare il rendering volumetrico su volumi scalari, bisogna applicare una serie di funzioni predefinite alle funzioni di trasferimento di opacità, colore e gradiente, mentre generalmente i metodi di visualizzazione per i diversi organi e tessuti sono preimpostati. Successivamente, a partire dall'immagine, l'utente deve estrarre delle ROI (regioni di interesse), le quali saranno visibili nel volume di ricostruzione 3D in direzione degli assi orizzontale, coronale e sagittale. Se si modifica l'intervallo che definisce una ROI si può vedere anche un cambiamento del volume ottenuto. Vi sono poi altre funzioni che possono essere utilizzate per avere una migliore visualizzazione possibile, come ad esempio la funzione soglia con cui è possibile regolare l'intensità dei colori in modo da distiguere meglio una zona da un'altra all'interno di un organo e in modo da definire meglio ad esempio il confine tra una lesione e il resto dei tessuti. [22] [23]

Infine, i risultati ottenuti possono essere salvati in formato STL o OBJ e possono essere sia sottoposti a stampa 3D o possono essere analizzati e visualizzati in altri software esterni.

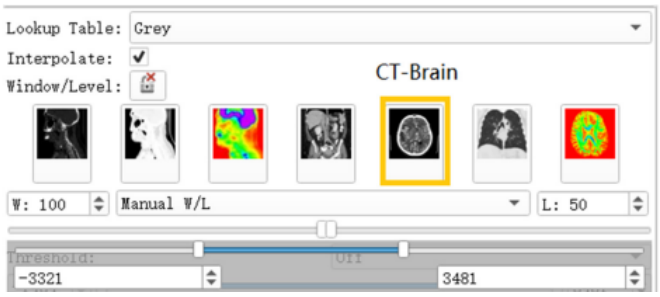

Figura 2.2. Esempi di scale di grigio usate comunemente in 3D Slicer.

2.2 Unity 3D

Unity 3D è un software opensource principalmente utilizzato per lo sviluppo di videogiochi, contenuti interattivi e più in generale ambienti 3D. Unity 3D è una piattaforma compatibile coi principali sistemi operativi Windows, Mac OS X e Linux, ed è disponibile con tre diverse tipologie di license, di cui una completamente gratuita (*Personal*), e due a pagamento (*Plus e Pro*). Esso deve inoltre la sua notorietà e il suo vasto utilizzo anche alla sua abilità di supportare un totale di 25 piattaforme, tra cui tutti i visori che utilizzano Open VR e Oculus SDK, smartphone (va bene sia per il sistema Android che iOS), console di gioco, browser come WebGL. Un altro punto a favore di Unity 3D è la possibilità di integrare molteplici pacchetti e librerie al suo interno che lo rendono ancor più versatile. Esso inoltre supporta come linguaggi di programmazione JavaScript e C#.

Oltre a ciò, si noti che è anche abbastanza semplice gestire i vari progetti che si vogliono creare su Unity 3D, in quanto grazie allo Unity Hub è possibile installare svariate versioni di Unity, tutte visibili nella voce **Installs**, e gestire i vari progetti che si sono creati, visibili invece alla voce **Projects**, come si può notare nella seguente immagine.

Una volta scelta la versione di Unity 3D per il progetto che si vuole creare, è visibile l'interfaccia grafica del software mediante la quale vengono gestite tutte le operazioni relative al progetto. Bisogna considerare anche l'eventualità di importare dall'esterno oggetti 3D già pronti precedentemente modellati con altri software come ad esempio Blender: in tal caso è necessario che essi siano in formato FBX, e qualora non lo fossero vanno convertiti nel formato richiesto. Inoltre durante l'importazione degli oggetti bisogna fare attenzione alle scale dimensionali che possono variare tra un software e l'altro. Si vede ora più nel dettaglio l'interfaccia grafica di Unity con i suoi diversi componenti.

1. **Hierarchy**: in questo pannello sono presenti gli oggetti presenti nella scena che si sta visualizzando, sia che essi siano oggetti 3D, sia che siano camere e

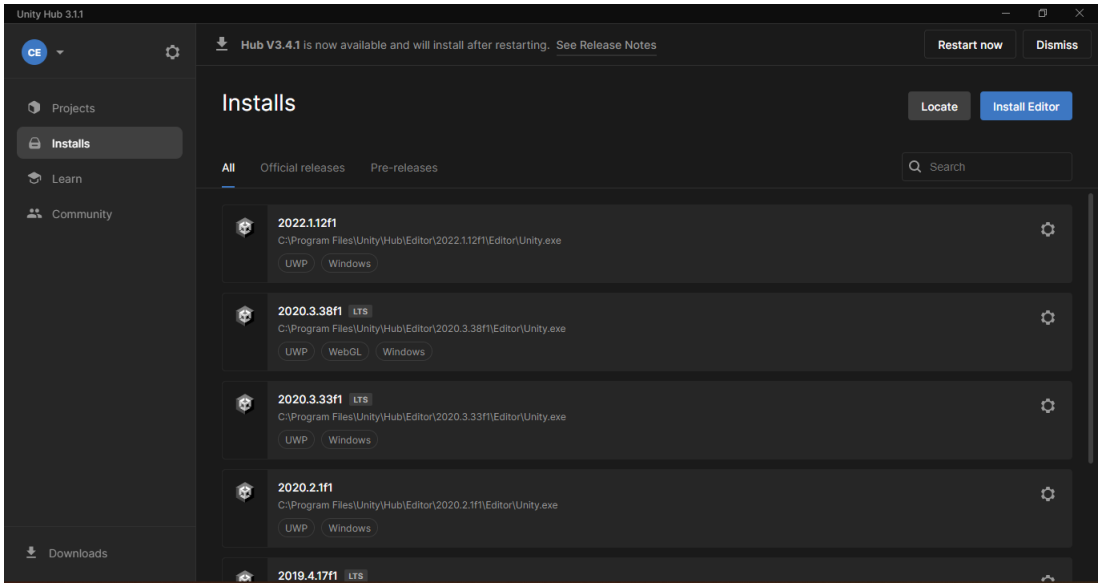

Figura 2.3. Unity Hub con diverse versioni del software installate.

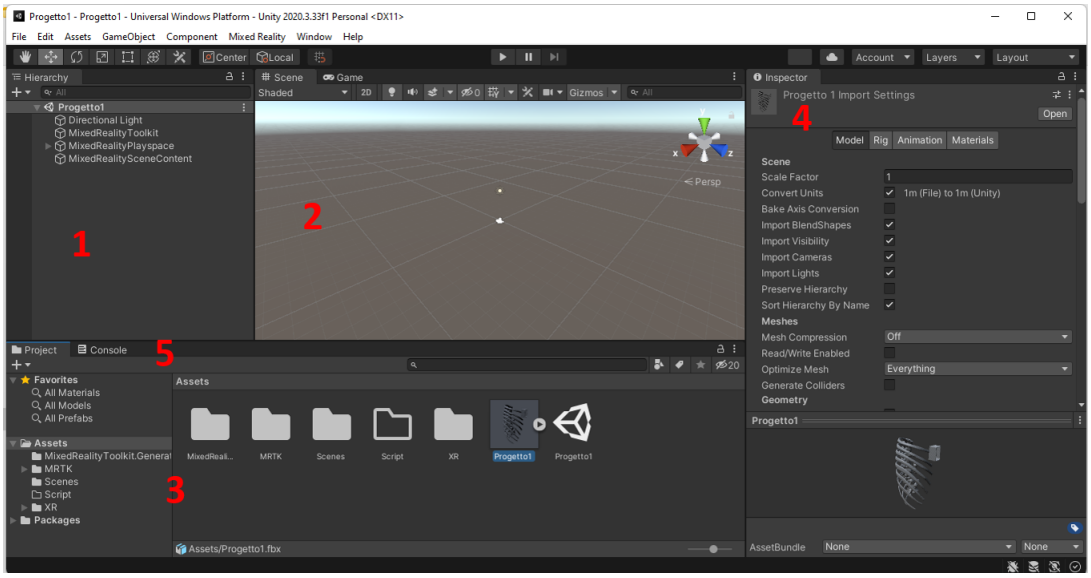

Figura 2.4. Interfaccia grafica di Unity con i suoi vari elementi evidenziati.

luci. Mediante questo pannello si possono creare, aggiungere e togliere nuovi elementi alla scena ed è possibile stabilire dei rapporti gerarchici tra essi. Un oggetto infatti può avere uno o più elementi figli, e se ad esempio si va

a interagire con un oggetto padre, si interagirà automaticamente anche con l'oggetto figlio.

2. **Scene**: è il pannello che mostra graficamente la scena con i suoi componenti, i quali possono essere manipolati e modificati al suo interno. Se si passa alla modalità play è possibile visualizzare una sorta di demo che mostra come effettivamente appare il progetto quando viene eseguito.

3. **Project**: qui sono presenti tutti gli asset che possono essere inclusi nel progetto, come ad esempio script associati ai vari elementi 3D, pacchetti ecc.

4. **Inspector**: esso mostra tutte le impostazioni e le proprietà riguardanti l'oggetto selezionato. Mediante questa finestra si possono effettuare tutte le modifiche volute sull'oggetto, e si può vedere se ad esso sono associati script con le loro relative variabili pubbliche che possono essere gestite mediante tale pannello.

5. **Console**: è la finestra mediante la quale è possibile visualizzare dei messaggi di errore, warning, avvisi generati durante l'esecuzione del progetto. Essa è fondamentale per testare la correttezza degli script associati ai vari elementi del progetto.

[24]

2.3 Mixed Reality Toolkit (MRTK)

Il Mixed Reality Toolkit è un insieme di pacchetti che permettono di sviluppare applicazioni multipiattaforma di realtà mista fornendo supporto ad hardware e ad altre piattaforme. Il toolkit si può intergrare all'interno del progetto di Unity mediante Unity Package Manager. Infatti, è possibile aggiungere i pacchetti MRTK nell'asset del progetto a partire dalla versione 2018.4 di Unity. Tra le principali funzioni del MRTK vi è quella di fornire un sistema di input multipiattaforma e blocchi predefiniti adibiti alla gestione delle interazioni spaziali. Inoltre, l'interfaccia utente permette la creazione rapida di prototipi, in quanto grazie a delle simulazioni è possibile visualizzare in tempo reale le modifiche effettuate. In aggiunta, il toolkit supporta svariati dispositivi come Microsoft Hololens 2 (utilizzato specificatamente in questo lavoro di tesi), Visori VR di Windows Mixed Reality, Android e iOS (tramite AR Foundation) grazie ai provider di plug-in per plug-in XR per Unity. Tuttavia, per quanto riguarda l'implementazione del Mixed Reality Toolkit all'interno di un progetto sviluppato in Unity, vi sono alcuni possibili contro: si ottengono infatti progetti di grandi dimensioni che richiedono tempi di compilazione più lunghi, le dipendenze dei pacchetti devono essere gestite manualmente così come gli aggiornamenti che possono richiedere passaggi multipli, e vi

è il rischio di perdere delle modifiche apportate al MRTK. I pacchetti disponibili sono:

- pacchetto **Foundation** (pacchetto di base): consente di sfruttare le funzionalità principali delle applicazioni di realtà mista;
- pacchetto **Extensions** (facoltativo): include funzionalità aggiuntive a quelle già fornite dal pacchetto Foundation;
- pacchetto **Tools** (facoltativo): include strumenti che aiutano a migliorare l'esperienza di sviluppo di realtà mista;
- pacchetto **Test Utilities** (facoltativo): utile per sviluppatori che creano componenti MRTK in quanto fornisce script utili per sviluppare dei test come ad esempio il test di simulazione manuale;
- pacchetto di esempi (facoltativo): contiene scene e script di esempio che simulano le funzionalità del pacchetto principale, contiene ad esempio demo di tracciamento oculare o input manuali.

Nel caso di sviluppo di applicazioni AR/VR è utile anche l'utilizzo del plug-in Mixed Reality Open XR, compatibile con le versioni di Unity a partire dalla 2020. Con l'inserimento di tale plug-in vengono importate nell'ambiente di Unity le funzionalità peculiari di Open XR, senza che il software sia coinvolto nel loro sviluppo. [25]

2.4 Microsoft Hololens 2

Questo lavoro di tesi è stato svolto mediante l'utilizzo del visore Hololens 2 prodotto dalla Microsoft. Tale dispositivo è molto performante e orientato principalmente verso lo sviluppo applicazioni di realtà mista, in cui vi è un'interazione in tempo reale tra gli oggetti 3D virtuali e il mondo reale circostante, a differenza della realtà aumentata in cui vi è solo una sovrapposizione di immagini virtuali al mondo reale. Dunque, la realtà mista si pone a metà strada tra la realtà aumentata e la realtà virtuale, e cerca di utilizzare gli aspetti più vantaggiosi di entrambe.

2.4.1 Caratteristiche principali di Hololens 2

Il visore Microsoft Hololens 2 è un dispositivo rilasciato a partire dal 2019. Esso è sostanzialmente un computer indossabile del peso di 566 grammi, è regolabile e può essere posto al di sopra di eventuali occhiali da vista, in quanto è costituito da un display con lenti olografiche trasparenti e munito di due sensori a raggi infrarossi in

grado di registrare i movimenti oculari dell'utente. Inoltre è un dispostivo che presenta ulteriori sensori: 4 telecamere a luce visibile per il tracciamento della testa, un sensore di profondità a tempo di volo da 1 MP, una IMU (Inertial Measurment Unit) cioè un sistema in cui sono integrati un accelerometro un magnetometro e un giroscopio, una videocamera con immagini da 8 MP e video 1080p30. Il processore utilizzato per l'elaborazione dei dati è il Qualcomm Snapdragon 850 e l'elaborazione olografica dei dati viene eseguita anche da una HPU (Holographic Process Unit) oltre che dalla GPU. Il dispositivo ha una connettività alla rete Wi-Fi, al Bluetooth ed è dotato di una memoria RAM di 8 GB. Tra le principali caratteristiche che lo rendono particolarmente performante vi sono la comprensione umana e la comprensione dell'ambiente. Infatti, oltre al tracciamento oculare già citato, Hololens 2 è in grado di effettuare anche il tracciamento di entrambe le mani, può essere gestito mediante comandi vocali grazie alla presenza di 5 canali audio, è in grado di effettuare il tracciamento dell'ambiente circostante a 6 gradi di libertà e di effettuare una mappatura spaziale in tempo reale, e infine è possibile effettuare foto e video, sia degli ambienti reali registrati che degli eventuali ologrammi e immagini virtuali presenti. Esso ha inoltre un'autonomia di 2-3 ore di utilizzo attivo. [26]

Figura 2.5. Microsoft Hololens 2.

Lo sviluppo di applicazioni per Hololens 2 è basato sull'utilizzo di UWP (Universal Windows Platform), su tale sistema infatti sono eseguibili la maggior parte delle applicazioni create per sistemi Microsoft, in quanto UWP è una piattaforma app comune per tutti i dispositivi che eseguono Windows 10. Infatti, se viene sviluppata un'applicazione che utilizza solo API di base, allora questa sarà eseguibile su qualunque dispositivo dotato di Windows 10 o di versioni successive, tuttavia talvolta si possono usare API specifiche del dispositivo, e in tal caso servono delle SDK di estensione. Nel capitolo successivo verrà descritto più nel dettaglio lo sviluppo di un'applicazione di realtà aumentata destinata al dispositivo Microsoft

Hololens 2, creata mediante l'utilizzo del software Unity 3D e successivamente compilata e distribuita sul visore tramite Visual Studio 2019.

Capitolo 3

Descrizione del progetto

Questo lavoro di tesi è stato svolto col fine di sviluppare un prototipo di un'applicazione distribuita su Microsoft Hololens 2 che permetta di effettuare una biopsia polmonare transtoracica, secondo una procedura basata sull'utilizzo della realtà aumentata. Come è stato già precedentemente dichiarato, l'AR è una tecnologia che negli ultimi anni è stata ampiamente investigata, ed è noto che possa apportare dei miglioramenti per quanto riguarda l'ottimizzazione di determinate procedure mediche. Nel caso di interventi di biopsia transtoracica, l'utilizzo di un dispositivo HDM, in cui l'utente possa vedere immagini 3D virtuali, consentirebbe di migliorare il flusso del lavoro grazie all'ottimizzazione dei tempi della procedura, inoltre il paziente verrebbe sottoposto a una minore dose di radiazioni e si avrebbe anche un minor tempo di utilizzo e manipolazione dell'ago durante l'intervento. Tuttavia, bisogna considerare un potenziale aspetto critico che si verificherebbe nell'effettuare un intervento utilizzando un dispositivo HDM in cui viene visualizzata la ricostruzione 3D del torace del paziente: è infatti di vitale importanza ottenere in un'applicazione del genere un perfetto matching spaziale tra il volume tridimensionale ricostruito e visualizzato sul display e il reale torace del paziente, affinchè l'inserimento dell'ago sia più preciso e sicuro possibile. In questo specifico lavoro si è cercato di ottenere un ancoraggio spaziale del modello 3D mediante la scansione di un QR code.

3.1 Procedura standard

La biopsia transtoracica è una procedura medica che viene praticata per effettuare un'indagine diagnostica su un paziente. Essa consiste nel prelievo di un campione di tessuto polmonare da un paziente che viene sottoposto ad anestesia locale, mediante l'utilizzo di un ago. L'obiettivo è quello di identificare eventuali anomalie nel tessuto, effettuare una diagnosi di probabili neoplasie e stabilire un piano di

cura per il paziente a seconda dei risultati ottenuti. La procedura standard di biopsia transtoracica avviene secondo diversi step. Infatti, il metodo tradizionale prevede l'acquisizione di diverse scansioni: la prima viene acquisita per identificare le coordinate esterne del sito di inserzione, le altre sono eseguite passo dopo passo durante l'inserimento dell'ago per valutare l'accuratezza della traiettoria, la distanza ago-bersaglio e il corretto raggiungimento della lesione. Quindi, il supporto delle immagini TAC è fondamentale per individuare il punto in cui va inserito l'ago e per guidare la traiettoria dell'ago fino al raggiungimento della lesione. Una volta che il bersaglio è stato raggiunto correttamente, con una siringa, si aspira il tessuto polmonare (frustolo) che verrà successivamente analizzato istologicamente da un anatomo-patologo. Durante tutta la procedura il paziente è sdraiatio sopra il lettino della TC o in posizione supina o prona, in base alla collocazione della lesione, inoltre deve eseguire in modo ppreciso le indicazioni del medico e deve rimanere il più fermo possibile.[27]

3.2 Procedura sperimentale

3.2.1 Ottenimento del volume

Il volume del modello è stato ottenuto utilizzando un fantoccio assemblato a partire dallo scheletro di emitorace stampato 3D e avvolto da diversi strati di gommapiuma. All'interno del fantoccio sono presenti delle palline disposte in modo casuale, che emulano delle ipotetiche lesioni o anomalie di tessuto che necessitano di essere investigate mediante la biopsia transtoracica ad ago.

Figura 3.1. Fantoccio stampato 3D ricoperto di gommapiuma.

Dunque, si tratta di un modello semplificato rispetto alla reale anatomia del torace di un essere umano, per cui in questo caso vengono tralasciati alcuni aspetti critici come la complessità delle strutture anatomiche e i movimenti degli organi interni. Quindi, al fine di ottenere la ricostruzione 3D del volume dell'emitorace, è stata effettuata una TC al suddetto fantoccio, e le immagini acquisite sono state salvate nel formato standard DICOM e poi sono state caricate nel software Slicer 3D.

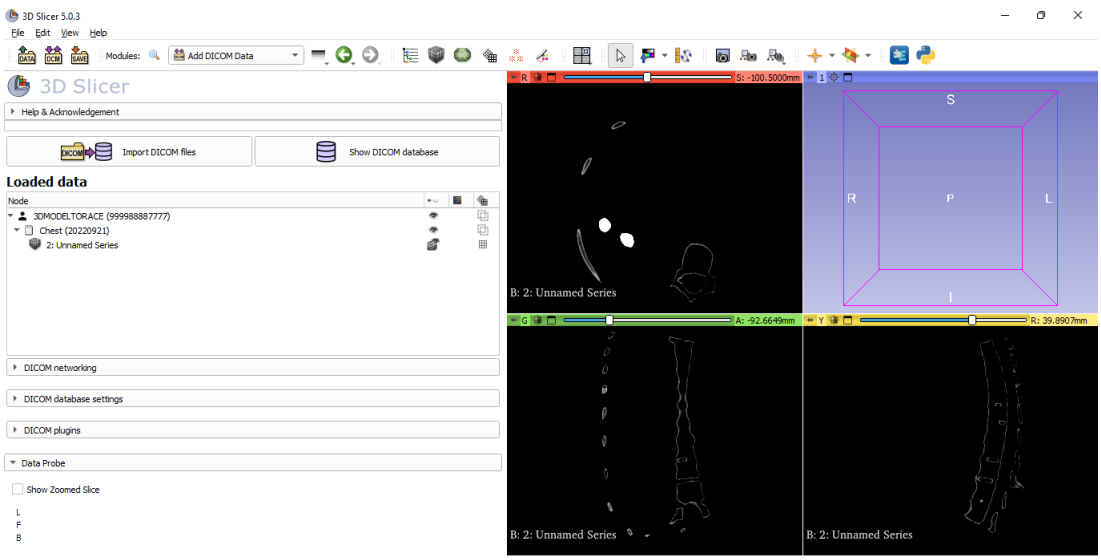

Figura 3.2. Immagini formato DICOM caricate su 3D Slicer.

Qui è stato effettuato il rendering volumetrico dopo aver leggermente lavorato sulle immagini effettuando una segmentazione delle stesse: prima, nella voce *Volumes* è stato selezionato il tipo di volume che si voleva ricostruire, poi sono stati impostati dei valori di soglia e si è lavorato attraverso il *Segment Editor*.

Infine, il file è stato esportato in formato STL (è possibile esportare il file anche in altri formati come ad esempio il formato OBJ). Nella seguente immagine è possibile visualizzare il volume ottenuto.

La maggior parte del lavoro è stata poi svolta sul software Unity 3D, in cui è stato caricato il modello 3D precedentemente ottenuto. Si noti che su Unity 3D non supporta il caricamento di file in formato STL ma in formato FBX, per cui la conversione dal formato STL ottenuto da Slicer 3D al formato FBX voluto da Unity è avvenuta utilizzando il software **Blender**. Blender è un software di progettazione 3D opensource e gratuito all'interno del quale sono integrate svariate funzionalità come la modellazione, il rendering, editing video, animazione, texturing e altro ancora. In questo caso Blender è stato usato esclusivamente per effettuare la

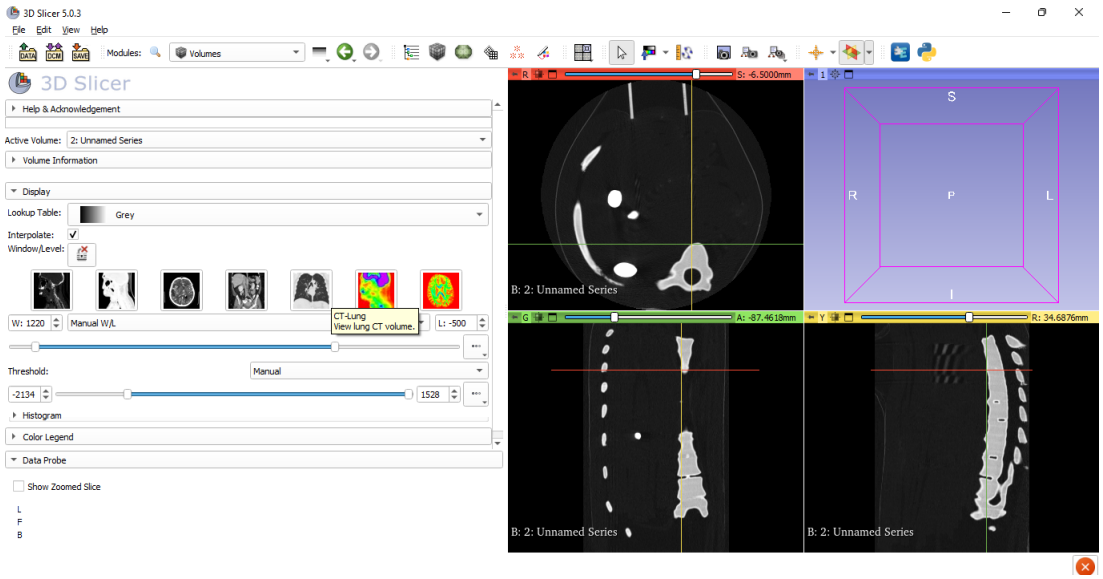

Figura 3.3. 3D Slicer Editor: selezione del tipo di volume.

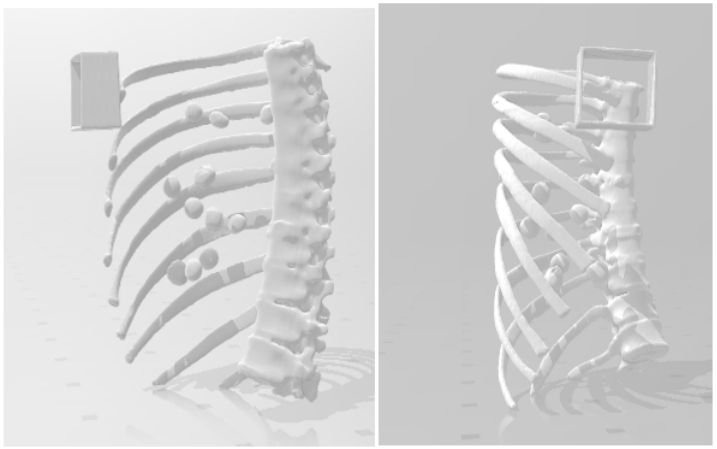

Figura 3.4. Volume ottenuto in formato STL.

conversione del modello nel formato desiderato e per spostare l'origine del modello all'altezza del cubo. Quest'ultima operazione servirà successivamente per ottenere un ancoraggio più preciso del QR code al modello, essendo il cubo il punto di riferimento per l'ancoraggio. Inoltre bisogna fare attenzione a mantenere invariata la scala del modello, quindi una volta che è stato importato su Unity, prima di metterlo in scena, bisogna impostare come valore di scala il valore 0.001 in quanto il modello risulta essere misurato in millimetri (il file STL ottenuto su 3D Slicer è

in millimetri) mentre Unity ha come Unità di misura di base il metro. Se non si impostasse il fattore di scala in modo corretto si otterrebbe un volume 1000 volte maggiore di quello iniziale. Dopo aver eseguito questi passaggi e dopo aver posto il modello con la scala giusta sulla scena, sull'inspector si vedrà che il valore di scala è impostato a 1, in tal caso si ha compatibilità tra le scale nei vari passaggi tra un software e l'altro. A questo punto si ha un volume 3D compatibile col software Unity 3D, in cui è avvenuta la restante parte del lavoro svolto. In alterativa su 3D Slicer si potrebbe salvare il modello anche come file OBJ, compatibile con Unity 3D, ma servirebbe comunque Blender per modificare l'origine del modello.

3.2.2 Ancoraggio del modello

L'obiettivo principale del progetto, come già dichiarato precedentemente, è quello di creare un prototipo in cui si riuscisse ad ancorare il modello 3D virtuale al fantoccio reale mediante la lettura di un codice a matrice. Pertanto, è importante capire dove posizionare il QR code nel torace del paziente e quanti QR code è opportuno apporre. Inoltre bisogna fare ulteriori considerazioni anche sulle dimensioni del QR code in questione. In questo caso si è scelto di apporre un solo QR code all'altezza del cubo visibile nel modello 3D che infatti costituisce una sorta di riferimento.

In generale il codice a matrice deve essere posizionato in un punto ben visibile, preferibilmente su una superficie piana, di facile accesso e che allo stesso tempo non limiti il lavoro che deve essere svolto. Quindi, osservando il modello, si può notare che è stato aggiunto un cubo, anch'esso stampato 3D insieme al resto del torace, con un lato di 5,5 centimetri ed è situato in corrispondenza della prima costola. Si vuole posizionare il codice a matrice sopra questo cubo. Nella seguente immagine è possibile visualizzare il modello importato su Unity. [28]

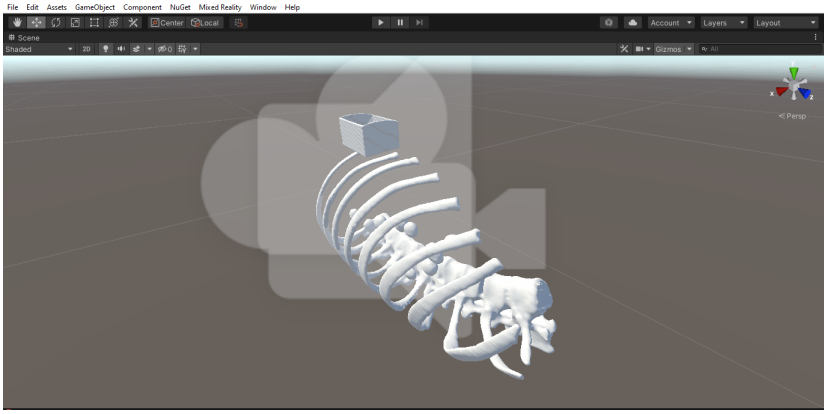

Figura 3.5. Modello 3D ottenuto su Unity .

Per quanto riguarda le caratteristiche del codice a matrice bisogna fare una serie di considerazioni: l'abilità di rilevare codici a matrice è influenzata anche dall'illuminazione, infatti bisogna avere sufficiente contrasto bianco/nero per avere un migliore rilevamento e si ha una prestazione migliore con dei codici a matrice opachi. Inoltre i dispositivi Windows Mixed Reality supportano le versioni di codice a matrice 1-10 e micro codici QR M1-M4, non hanno l'abilità di rilevare codici minori di 5 cm di lato e, per codici con lato che va dai 5 cm ai 10 cm, la capacità di rilevamento è influenzata dalla disanza in cui ci si trova rispetto al codice. Complessivamente si possono avere QR code di lato che va dai 5 cm ai 25 cm e la distanza per detectarlo può andare dai 50 cm ai 2 m dal codice più piccolo al più grande. In questo specifico caso quindi è stato scelto un codice di 5,5 cm di lato, in modo che sia ben posizionato sopra al cubo di riferimento, ipotizzando una distanza per il rilevamento che possa andare dai 50 cm ai 70 cm (distanza tra il paziente steso sul lettino e il medico che indossa il visore in grado di scannerizzare il QR code). Nel rilevare il QR code col visore può essere necessario avvicinarsi leggermente al cubo.

Figura 3.6. QR code utilizzato.

Tale codice a matrice è stato generato utilizzando un generatore di QR code online. Una volta fatte tutte le considerazioni opportune sulla scelta del QR si è implementato il tutto grazie all'utilizzo del software Unity 3D.

3.2.3 Implementazione del progetto su Unity 3D

Per sviuppare un'applicazione di realtà aumentata su Unity, il primo step fondamentale è costituito dalla configurazione del software per lo sviluppo con Windows Mixed Reality (in questo specifico caso è stata utilizzata la versione di Unity

2020.2.38). Infatti, una volta creato un nuovo progetto 3D, il primo step fondamentale è scaricare il pacchetto MRTK e aggiungerlo a una scena Unity. Per fare ciò è stato eseguito il file MixedRealityTool.exe, e sono stati selezionati Mixed Reality Toolkit Foundation e Mixed Reality OpenXR Plugin. Quindi, una volta importate queste funzionalità, viene riavviato il progetto e configurato con il toolkit di realtà mista e, all'interno del MRTK, viene abilitata la gestione dei plug-in XR per aggiungerlo al progetto. Successivamente si seleziona Project Settings sull'Editor di Unity e si verifica che tutti i parametri desiderati siano impostati come si può osservare nella seguente figura.

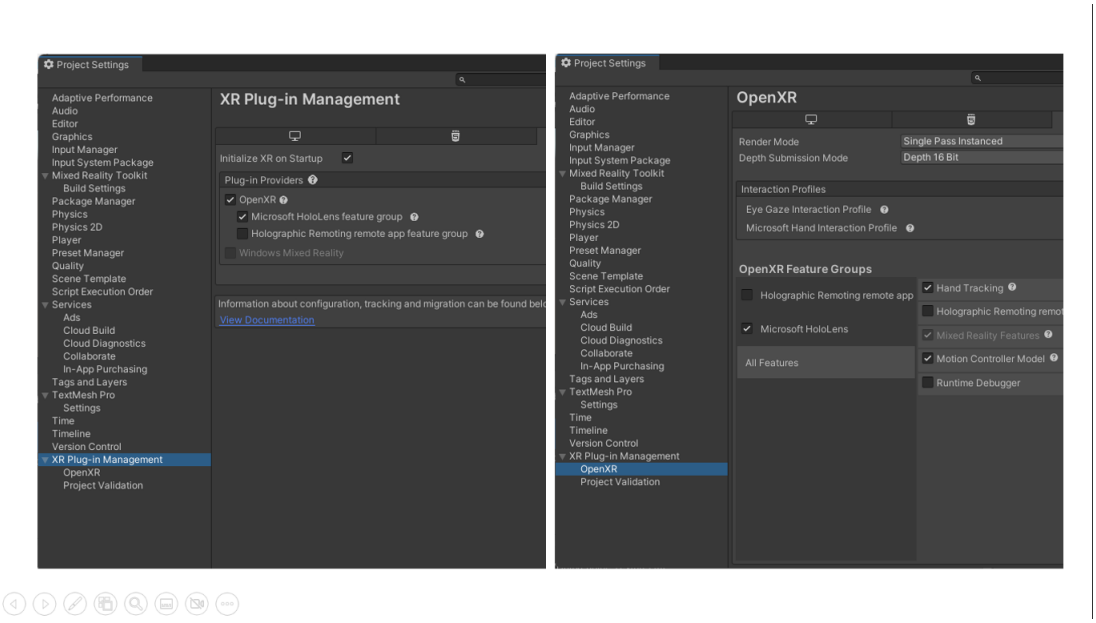

Figura 3.7. Parametri impostati su Project Settings.

A questo punto il progetto è stato configurato correttamente con il Mixed Reality Toolkit. Tuttavia è necessario aggiungere altri pacchetti: innanzitutto bisogna aggiungere il pacchetto NuGet per Unity, poi da qui è stato scaricato il pacchetto Microsoft.MixedReality.QR, quindi si può verificare la corretta installazione dei pacchetti andando in Asset, poi su Packages e qui si può verificare se essi sono presenti.

Una volta configurato correttamente il progetto si possono aggiunegre tutti gli altri elementi necessari alla costruzione dell'applicazione: sulla scena infatti sono presenti una serie di game object oltre il modello del torace importato da Blender e ad essi sono associati vari script che consentono la corretta esecuzione del progetto. Infatti, sulla scena che è stata costruita, è presente un pannello con due pulsanti, uno finalizzato a dare inizio alla scansione del QR, l'altro a terminarla.

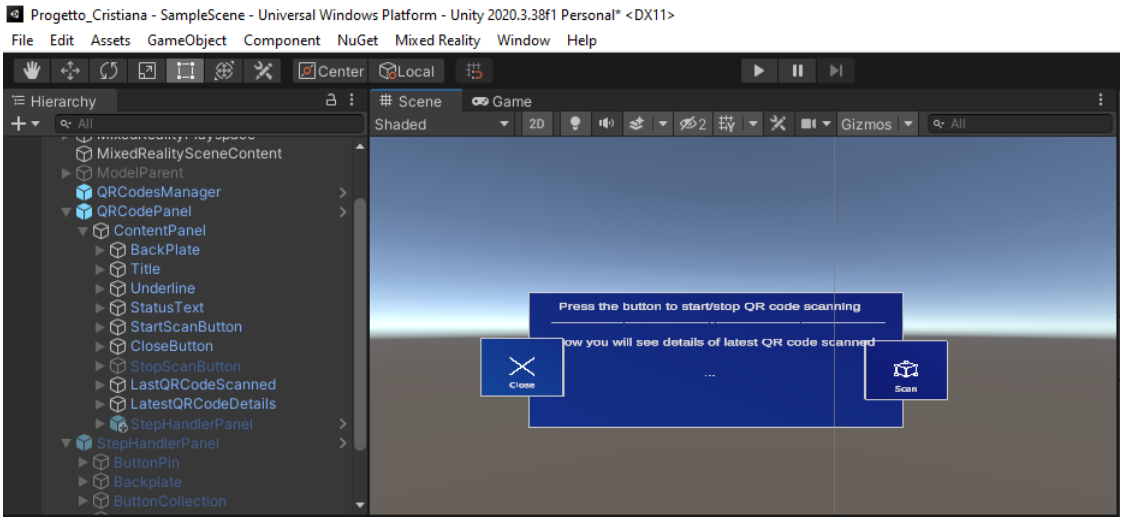

Figura 3.8. QRCode Panel visto in Hierarchy e Scene.

Nella figura infatti si possono osservare sulla Hierarchy tutti gli elementi che costituiscono il game object QRCodePanel mentre nella scena si vede com'è costruito il panel e i due pulsanti di Start Scan (che durante l'esecuzione dell'app si tramuta in Stop Scan) e Close.

Il funzionamento di base dell'applicazione è il seguente: l'utente avvia l'applicazione sul visore, come prima cosa visualizza il panel e quando vuole dare origine alla scansione preme il tasto Scan. Durante la fase di scansione l'applicazione è in grado di riconoscere solo ed esclusivamente il QR code desiderato, per cui si avrà un corretto funzionamento solo se verrà aggangiato il QR code voluto. In caso affermativo, una volta scansionato il QR code specifico, premendo il tasto Stop, la ricerca del QR termina; all'altezza esatta del QR code comparirà il modello del torace, inizialmente non visibile. Nel panel compaiono anche le coordinate spaziali rispettivamente del modello e del QR code. In caso contrario, se si inquadrasse un QR code diverso da quello voluto, una volta premuto il tasto Stop, comparirebbe la seguente scritta: "QR code not found. Please start scan again!". L'attività del panel è gestita principalmente dal game object QRCodeManager a cui sono collegati due script: QR Code Manager e QR Code Visualizer.

Come si può osservare dall'Inspector del QRCodeManager, allo script QR Code Visualizer bisogna associare dei parametri: il modello del torace che deve comparire una volta effettuata la scannerizzazione, un QRCodePrefab che comparirà sopra al QR code voluto, e degli elementi di testo che compaiono durante lo svolgimento della scansione. Prima di associare il modello allo script, esso viene reso figlio di un altro game object vuoto: questa operazione serve a bloccare la rotazione del

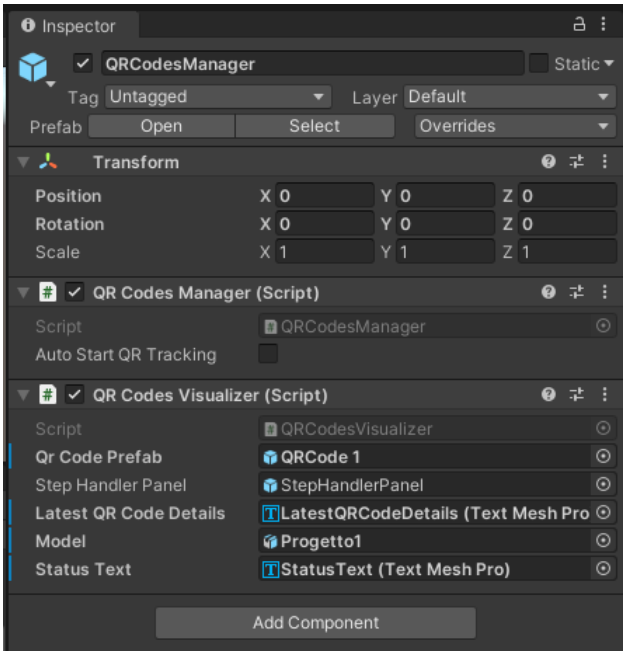

Figura 3.9. QR Code Manager visto dall'Inspector.

modello e a definire invece una posizione che lo renda non visibile quando viene avviata l'applicazione. Quindi, nello specifico allo script viene associato il genitore del modello.

Inoltre all'interno dello script QR Code Visualizer viene dichiarato il nome del codice QR che si desidera cercare, affinchè ci sia l'abilità di far comparire il modello solo quando viene agganciato quello specifico codice a matrice. Il modello si posiziona sul QR code all'altezza del cubo che costituiva il punto di riferimento per l'apposizione del QR code cartaceo reale, e viene posizionato orizzontalmente. Tale orientamento orizzontale del modello è preferibile in quanto si ipotizza di svolgere la scansione con un paziente sdraiato con il QR code posto sul torace in corrispondenza del cubo di riferimento. A seconda dell'orientameno del QR code si ottiene una posizione del modello leggermente diversa, tuttavia non si ha mai una rotazione del modello.

Dunque, una volta conclusa la realizzazione del progetto su Unity, bisogna impostare tutti i parametri necessari per compilare e distribuire l'applicazione nel dispositivo Microsoft Hololens 2. Per fare ciò, bisogna andare alla voce Built Settings e impostrare come piattaforma Universal Windows Platform, qui bisogna selezionare i parametri relativi al dispositivo target, come la versione di SDK desiderata, il tipo di architettura, la versione della piattaforma ecc. Tutti questi parametri sono visibili nella seguente immagine.

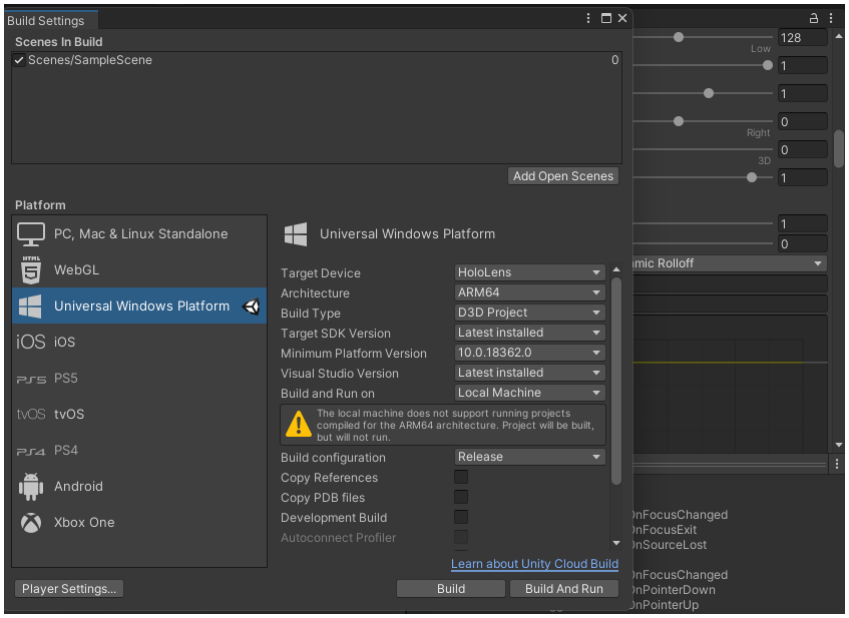

Figura 3.10. Parametri impostati in Built Settings.

Infine, una volta effettuato il cambio della piattaforma si procede con la Built dell'applicazione. Alla fine della compilazione su Unity si ottengono una serie di file; la compilazione e la distribuzione dell'applicazione su Microsoft Hololens 2 avvengono mediante Visual Studio 2019. In questa fase entra in gioco anche il dispositivo Hololens 2, in quanto è necessario associare il computer da cui si effettua la distribuzione al visore. Pertanto entrambi i dispositivi sono stati collegati alla stessa rete Wi-Fi, poi viene inserito l'indirizzo IP, rilevato sul visore, all'interno del progetto su Visual Studio, e infine si avvia la compilazione in modalità Release su Visual Studio e si seleziona l'architettura ARM64. Inoltre, terminata la compilazione, viene richiesto l'inserimento di un PIN che si può vedere tra le impostazioni del visore e serve per associare i due dispositivi.

Quindi, una volta effettuato l'accoppiamento e terminata la compilazione su Visual Studio, si avrà la distribuzione dell'applicazione su Hololens 2. In media, compilazione e distribuzione avvengono in circa 7 minuti.

Di seguito sono presenti delle immagini che mostrano quanto si vede dal visore. Nella prima (figura 3.12) si vede cosa compare sul pannello una volta scannerizzato un QR code diverso da quello desiderato. Nella figura 3.13 si vede come compare il modello virtuale in corrispondenza del fantoccio, e infine (figura 3.14) si noti il pannello con le informazioni relative al QR code desiderato, che in questo caso era chiamato "Patient".

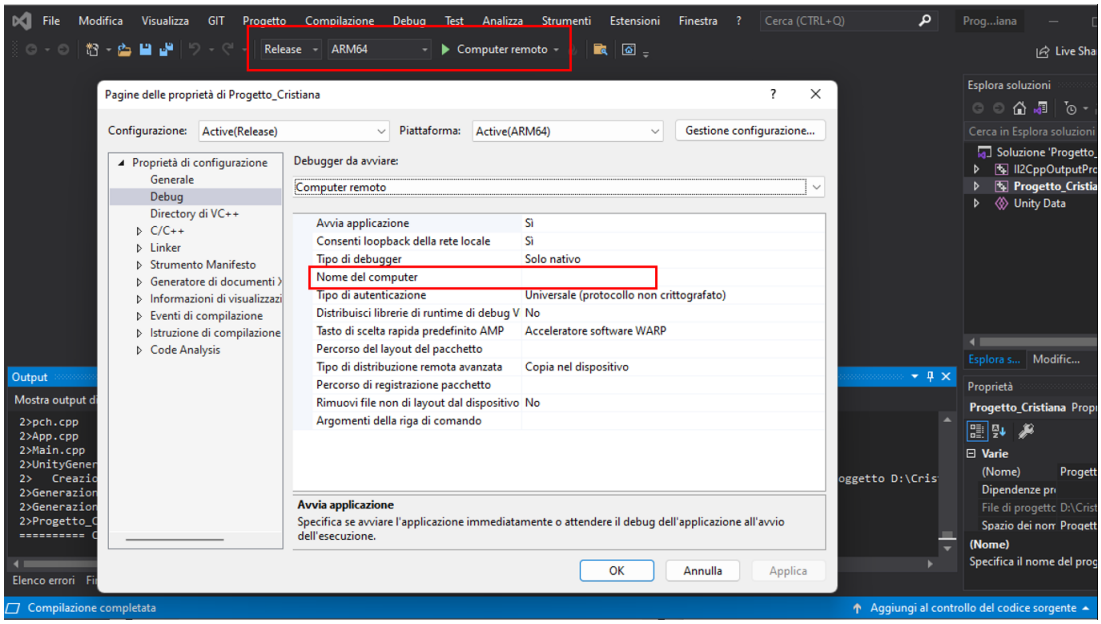

Figura 3.11. Parametri imposti per compilazione e distribuzione su Visual Studio 2019.

Si vede che ogni volta che viene effettuata una scannerizzazione vengono restituite le cordinate spaziali del modello e del QR code. Quando avviene l'aggancio del modello mediante la scansione del QR, potrebbe accadere di avere un leggero disallineamento tra la gabbia toracica virtuale e il fantoccio reale, pertanto potrebbe essere necessaria una leggera correzione manuale dell'orientamento del fantoccio. Ciò avviene poichè in fase di ancoraggio il matching spaziale si ottiene in corrispondenza del cubo, per cui non si ha un controllo totale sulla posizione della restante parte della gabbia toracica.

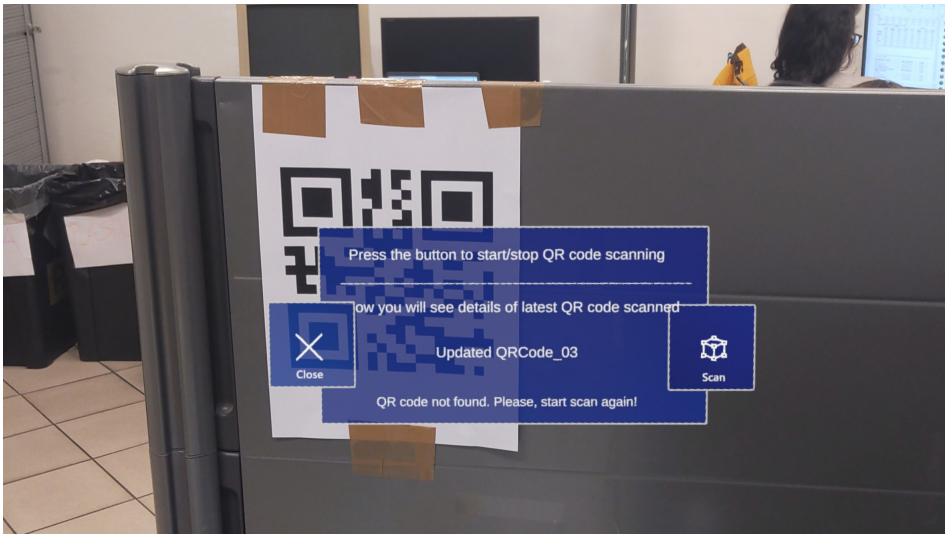

Figura 3.12. Pannello dopo aver scannerizzato QR code non desiderato.

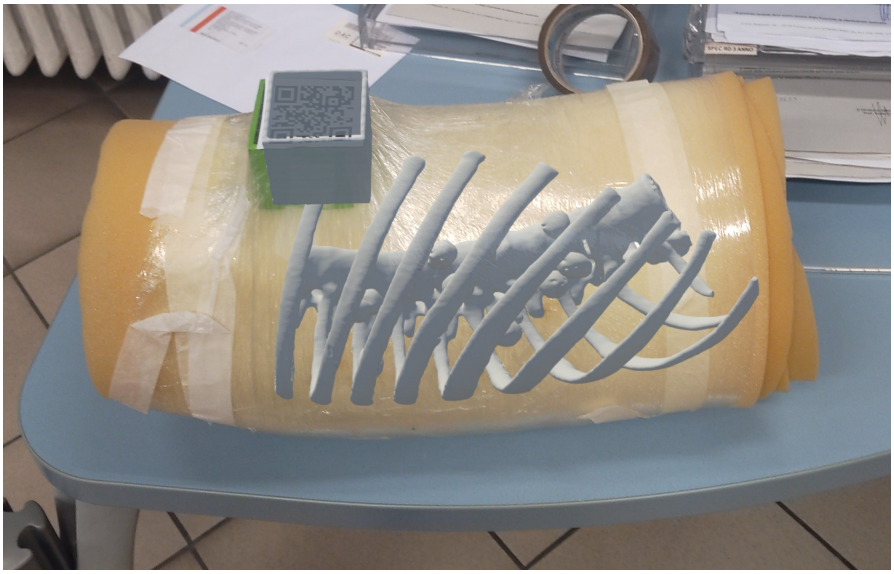

Figura 3.13. Corrispondenza tra modello reale e virtuale.

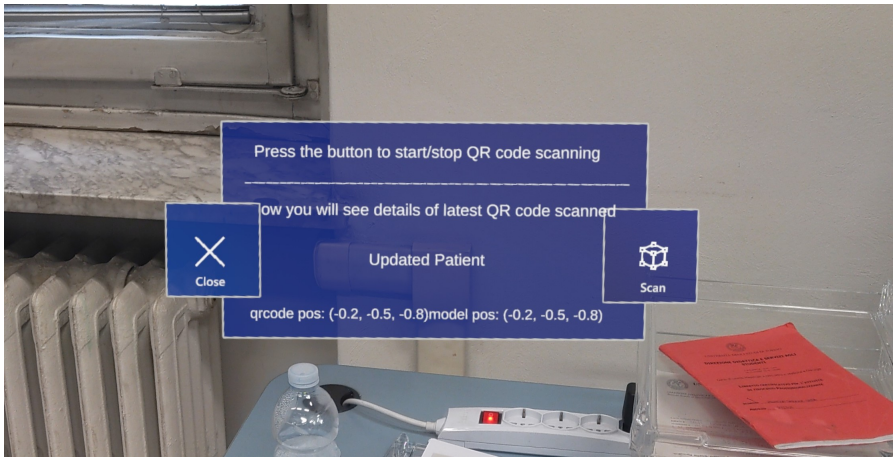

Figura 3.14. Pannello con le informazioni relative al QR code voluto.

Capitolo 4

Risultati ottenuti

Il modello presenta dieci palline che simulano delle ipotetiche lesioni (ad esempio metastasi) del polmone, ed esse sono collocate in modo casuale all'interno dell'emitorace per cui può essere più o meno difficile arrivare ad una lesione anche a seconda della sua collocazione. Tali difficoltà si riscontrano anche nella realtà e possono avere delle ripercussioni sulla durata della procedura. Inoltre nel caso di un paziente reale bisogna considerare la presenza di altri tessuti molli e vasi sanguigni per cui si ha una situazione di partenza già più complessa.

L'applicazione creata è stata quindi testata presso l'ospedale Molinette di Torino utilizzando il fantoccio e provado a inserire al suo interno un ago da biopsia, quindi è stata effettuata una simulazione dell'intera procedura in sala TC.

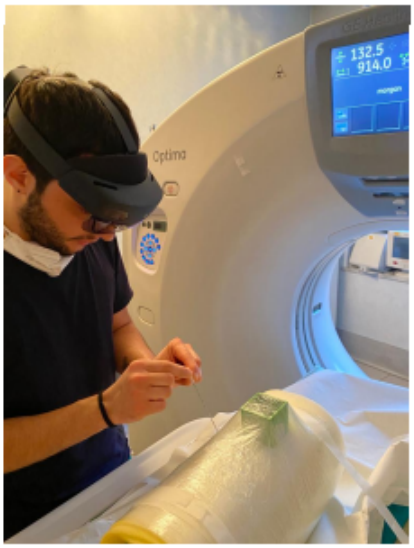

Figura 4.1. Simulazione procedura sperimentale.

Il test è stato svolto da tre medici con esperienze diverse: vi è infatti un utente esperto, uno con esperienza intermedia e uno con un'esperienza da principiante. Essi hanno svolto la procedura sia seguendo il metodo tradizionale che quello sperimentale proposto ed è stato effettuato un confronto quantitativo tra i due, analizzando diversi parametri:

- il **tempo** impiegato per svolgere la procedura;
- il **numero di scansioni** richiesto per completare con successo la procedura, esclusa la scansione iniziale;
- la **correttezza** che è definita come la percentuale di prossimità raggiunta con l'ago rispetto al nucleo della lesione. Con questo parametro si cerca di valutare l'accuratezza della procedura.
- il **tempo risparmiato** nell'esecuzione della procedura sperimentale rispetto a quella tradizionale;
- la **percentuale di tempo risparmiato** tra le due procedure.

Di seguito sono riportati in maniera schematica i risultati ottenuti.

		Metodo tradizionale			Metodo sperimentale			Risultati	
		Tempo	Scans	Correttezza	Tempo	Scans	Correttezza	Tempo risparmiato	Percentuale di tempo risparmiato
Utente 1 (Esperto)	**Target facile**	05:34	3	95%	03:15	1	98%	02:19	42%
	Target intermedio	06:11	3	83%	03:30	1	95%	02:41	43%
	Target difficile	06:41	4	78%	03:50	2	95%	02:51	43%
Utente 2 (Esperienza intermedia)	**Target facile**	06:50	3	95%	04:00	1	98%	02:50	41%
	Target intermedio	07:00	4	80%	04:09	1	93%	02:51	40%
	Target difficile	08:39	4	75%	04:20	2	90%	03:19	43%
User 3 (Principiante)	**Target facile**	07:45	4	88%	04:20	1	97%	03:25	43%
	Target intermedio	08:38	4	70%	04:56	2	88%	03:42	43%
	Target difficile	11:02	5	65%	05:25	2	80%	05:37	51%

Figura 4.2. Risultati ottenuti: confronto tra i due metodi.

Si noti come l'applicazione proposta riduce notevolmente il numero di scansioni a cui dev'essere sottoposto un paziente, permettendo di diminuire anche la dose di radiazioni a cui egli dovrebbe essere soggetto.

Inoltre, con l'esecuzione della procedura sperimentale, si ottiene un notevole risparmio di tempo: infatti in media la procedura sperimentale viene eseguita in 4,2 minuti mentre col metodo tradizionale si ha una durata media di 7,6 minuti. Ciò significa che la messa a punto di un sistema come quello proposto consentirebbe un notevole aumento di produttività all'interno di un reparto, e nel corso di una giornata lavorativa sarebbe possibile eseguire un maggior numero di biopsie percutanee rispetto a quanto avviene col metodo tradizionale.

Infine, grazie al confronto tra le due procedure, emerge un altro dato interessante: si vede infatti che il valore percentuale di correttezza è aumentato nel caso del metodo sperimentale e la disparità di correttezza percentuale tra un utente esperto e un principiante è minore. Ciò indica che tale metodo sperimentale eliminerebbe in parte anche la soggettività che caratterizza il successo di una procedura eseguita col metodo tradizionale, in quanto, in quest'ultimo caso, l'abilità dell'utente ha un maggiore rilievo.

4.1 Analisi del rischio

L'utilizzo della realtà aumentata in ambito medico comporta la valutazione di una serie di aspetti potenzialmente critici, primo fra tutti il comfort dell'utente durante l'utilizzo del visore. Infatti, come è stato già precedentemente accennato, l'uso prolungato del visore può non essere confortevole per il medico che deve eseguire la procedura e può provocare sintomi come mal di testa e nausea.

Inoltre, per quanto riguarda questa specifica applicazione, bisogna considerare che in tal caso è stato utilizzato un fantoccio stampato 3D, per cui si ha un modello estremamente semplificato rispetto a ciò che si avrebbe nella realtà. La ricostruzione di un volume anatomico reale può essere molto più complessa e lunga, in quanto non possono essere trascurati altri organi e tessuti, e bisogna anche gestire una maggiore mole di dati.

Un altro aspetto critico è la necessità di dover correggere l'allineamento del modello rispetto al fantoccio reale, poichè si ha un ancoraggio del QR code all'altezza del cubo di riferimento, ma servirebbe comunque una maggiore precisione per quanto riguarda la gestione della restante parte del modello.

In aggiunta, per poter utilizzare su esseri umani un'applicazione simile, è necessario che ci sia una precisione al millimetro per quanto riguarda il matching spaziale tra modello virtuale e reale; a tal proposito, un aspetto fondamentale è anche la necessità che il QR code rimanga più fermo possibile. Sul fantoccio è

possibile ottenere un QR code immobile, ma passando a un paziente reale bisogna tenere conto di eventuali movimenti involontari dati dalla respirazione e dai movimenti degli organi interni del paziente stesso. Ciò comporterebbe un maggior rischio di perdita di ancoraggio involontaria o una minore precisione per quanto riguarda il matching spaziale desiderato.

4.2 Sviluppi futuri

L'applicazione descritta in questo lavoro di tesi rappresenta un prototipo di come potrebbe svolgersi una biopsia transcutanea polmonare mediante l'utilizzo della realtà aumentata.

Il lavoro svolto costituisce una base su cui possono essere implementate altre funzionalità che ne aumentino la precisione e la completezza. In particolare, si è detto che un problema è costituito da un imperfetto allineamento tra il modello reale e virtuale, nonostante l'ancoraggio al QR code avvenga con successo. A tal proposito si potrebbe pensare di integrare delle gesture al modello, cioè si potrebbe conferire al modello l'abilità di essere spostato manualmente dall'utente, il quale potrebbe ruotarlo e spostarlo nello spazio con una o entrambe le mani in modo abbastanza intuitivo. In alternativa o in aggiunta alle gesture si potrebbero integrare nell'applicazione anche dei comandi vocali specifici. Tuttavia, ciò potrebbe non garantire comunque l'ottenimento di un allineamento estremamente preciso (al millimetro) tra i due modelli.

Un'altra possibilità potrebbe essere l'aggiunta di alcuni tasti sull'interfaccia che consentano di regolare il modello lungo gli assi sia per quanto riguarda la rotazione sia per quanto riguarda la posizione. In questo modo si avrebbe un controllo più fine sulla posizione del modello virtuale rispetto a quanto potrebbe avvenire con l'implementazione delle gesture.

Inoltre, sempre al fine di ottenere un matching spaziale quanto più preciso possibile, si potrebbero introdurre altri QR code da leggere e ancorare. Infatti, si è visto che l'applicazione in questione funziona attraverso la lettura di un QR code desiderato e posizionato all'altezza del cubo, ma un'alternativa potrebbe essere costituita dall'individuare dei punti di repere sul torace del paziente ed ivi sistemare altri QR code. Si può pensare ad esempio a una configurazione con 3 QR code (i punti di repere in cui appore i QR code dovrebbero essere quanto più piani e fermi possibile durante la procedura), come quella presente nella seguente immagine, in cui le X rosse rappresentano i possibili punti scelti. Inoltre i QR code posizionati non dovrebbero essere d'intralcio per tutta l'esecuzione della procedura, in particolare per l'inserimento dell'ago.

Per quanto riguarda invece la completezza dell'applicazione, si è visto che è stato necessario effettuare dei passaggi tra vari software e che si è dovuta prestare

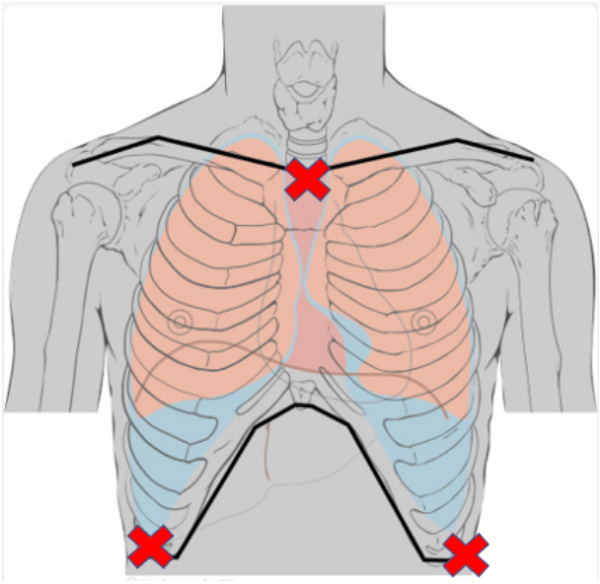

Figura 4.3. Possibile configurazione con 3 QR code.

una particolare attenzione alle scale dimensionali del modello. Nella realtà svolgere continuamente tali passaggi potrebbe risultare macchinoso e allungherebbe le tempistiche, per cui si potrebbe pensare di automatizzare la conversione tra il formato STL e il formato FBX, magari attraverso l'utilizzo di uno script integrato su Unity o di un plug-in.

Capitolo 5

Conclusioni

Il progetto descritto in questo lavoro di tesi dimostra come un sistema AR integrato nel dispositivo Microsoft Hololens 2 riduca le tempistiche di esecuzione della procedura e il numero di scansioni TC necessarie. Si è visto anche che esso favorisce un'accurata localizzazione delle lesioni.

D'altra parte si è visto anche che vi è un ampio margine di miglioramento del sistema, specialmente per quanto riguarda la riduzione delle tempistiche di elaborazione dell'immagine e della catena che porta dal file formato DICOM all'ottenimento dell'ologramma. Infatti le prestazioni generali del software potrebbero migliorare notevolmente ottenendo una più immediata visualizzazione del modello 3D all'interno del software, soprattutto se si pensa ad un potenziale utilizzo in ampia scala su pazienti reali.

Tra i progressi futuri, come è stato già precedentemente accennato, sempre nell'ottica di un utilizzo su pazienti reali, vi sono lo sviluppo e l'integrazione di maggiori dettagli realistici del modello, come ad esempio dettagli riguardandi l'atto respiratorio di un paziente, ma anche l'ottenimento di un'immediata fuzione tra ologramma e bersaglio fisico.

Tuttavia, sebbene il prototipo realizzato presenti notevoli semplificazioni rispetto a quanto si avrebbe nella realtà, è comunque evidente che un sistema del genere possa essere significativo anche nella formazione del personale medico. Infatti il dato emerso è che tale sistema potrebbe accorciare la distanza tra un medico esperto e un tirocinante, per cui esso può essere considerato uno strumento in grado di migliorare le prestazioni e la competenza del nuovo personale medico.

Ciò dimostra come un sistema AR ad uso medicale possa essere versatile, in quanto potrebbe garantire contemporaneamente delle migliori prestazioni dal punto di vista diagnostico e della produttività, ma anche dal punto di vista della formazione del personale medico. Pertanto, tali vantaggi giustificano l'elevata diffusione di studi clinici sviluppati negli ultimi anni e incentrati sull'implementazione di sistemi AR ad uso medicale. Infatti, sebbene non siano poche le difficoltà e i

costi che bisogna fronteggiare nel caso in cui si voglia sviluppare un sistema AR medicale, vi è comunque una forte volontà da parte della comunità scientifica di superare tali criticità dati i benifici che se ne trarrebbero.

Ringraziamenti

Ringrazio le professoresse Gabriella Balestra, Samanta Rosati e Valentina Giannini per avermi dato la possibilità di svolgere questo lavoro di tesi. In particolare ringrazio Valentina Giannini per professionalità e la disponibilità con cui mi ha sempre seguito. Ringrazio inoltre Giovanni Maimone per il supporto che mi ha fornito, e il dottor Andrea Di Chio dell'ospedale Molinette di Torino per la passione con cui ha seguito il progetto.

Bibliografia

[1] Ronald T. Azuma. A survey of augmented reality. *Presence: Teleoperators and Virtual Environments (1997)*, 6(4):355–385, 1997.

[2] Jens Grubert, Tobias Langlotz, Stefanie Zollmann, and Holger Regenbrecht. Towards pervasive augmented reality: Context-awareness in augmented reality. *IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER GRAPHICS)*, 23(6):1706 – 1724, 2017.

[3] Yunqiang Chen, Qing Wang, Hong Chen, Xiaoyu Song, Hui Tang, and Mengxiao Tian. An overview of augmented reality technology. *Journal of Physics: Conference Series*, 1237(2):022082, jun 2019.

[4] G. Dini and M. Dalle Mura. Application of augmented reality techniques in through-life engineering services. *Procedia CIRP*, 38:14–23, 2015. Proceedings of the 4th International Conference on Through-life Engineering Services.

[5] Vassilios Vlahakis, M Ioannidis, John Karigiannis, Manolis Tsotros, Michael Gounaris, Didier Stricker, Tim Gleue, Patrick Daehne, and Luis Almeida. Archeoguide: an augmented reality guide for archaeological sites. *IEEE Computer Graphics and Applications*, 22(5):52–60, 2002.

[6] Mehdi Mekni and André Lemieux. Augmented reality : Applications , challenges and future trends. 2014.

[7] Wee Sim Khor, Benjamin Baker, Kavit Amin, Adrian Chan, Ketan Patel, and Jason Wong. Augmented and virtual reality in surgery—the digital surgical environment: applications, limitations and legal pitfalls. *Annals of translational medicine*, 4(23), 2016.

[8] Jan Fischer, Markus Neff, Dirk Freudenstein, and Dirk Bartz. Medical augmented reality based on commercial image guided surgery. In *EGVE*, pages 83–86, 2004.

[9] Ho-Gun Ha and Jaesung Hong. Augmented reality in medicine. *Hanyang Medical Reviews*, 36(4):242–247, 2016.

[10] Raul N Uppot, Benjamin Laguna, Colin J McCarthy, Gianluca De Novi, Andrew Phelps, Eliot Siegel, and Jesse Courtier. Implementing virtual and augmented reality tools for radiology education and training, communication, and clinical care. *Radiology*, 291(3):570–580, 2019.

[11] Ci-Jyun Liang, Charles Start, Hanna Boley, Vineet R Kamat, Carol C Menassa, and Michelle Aebersold. Enhancing stroke assessment simulation experience in clinical training using augmented reality. *Virtual Reality*, 25(3):575–584, 2021.

[12] Petr Vávra, Jan Roman, Pavel Zonča, Peter Ihnát, Martin Němec, Jayant Kumar, Nagy Habib, and Ahmed El-Gendi. Recent development of augmented reality in surgery: a review. *Journal of healthcare engineering*, 2017, 2017.

[13] Naoki Suzuki, Asaki Hattori, and Makoto Hashizume. Benefits of augmented reality function for laparoscopic and endoscopic surgical robot systems. *navigation*, 1:6, 2008.

[14] Mohammed Ahmed Abdelrazek Mohammed, Mohamed H Khalaf, Andrew Kesselman, David S Wang, and Nishita Kothary. A role for virtual reality in planning endovascular procedures. *Journal of Vascular and Interventional Radiology*, 29(7):971–974, 2018.

[15] Brian J Park, Stephen J Hunt, Charles Martin III, Gregory J Nadolski, Bradford J Wood, and Terence P Gade. Augmented and mixed reality: technologies for enhancing the future of ir. *Journal of Vascular and Interventional Radiology*, 31(7):1074–1082, 2020.

[16] Fangyi Liu, Zhigang Cheng, Zhiyu Han, Xiaoling Yu, Mingan Yu, and Ping Liang. A three-dimensional visualization preoperative treatment planning system for microwave ablation in liver cancer: a simulated experimental study. *Abdominal Radiology*, 42:1788–1793, 2017.

[17] David B Douglas, Demetri Venets, Cliff Wilke, David Gibson, Lance Liotta, Emanuel Petricoin, Buddy Beck, and Robert Douglas. Augmented reality and virtual reality: Initial successes in diagnostic radiology. *State of the art virtual reality and augmented reality knowhow*, 2018.

[18] David B Douglas, Clifford A Wilke, J David Gibson, John M Boone, and Max Wintermark. Augmented reality: Advances in diagnostic imaging. *Multimodal Technologies and Interaction*, 1(4):29, 2017.

[19] Mohammad Elsayed, Nadja Kadom, Comeron Ghobadi, Benjamin Strauss, Omran Al Dandan, Abhimanyu Aggarwal, Yoshimi Anzai, Brent Griffith,

Frances Lazarow, Christopher M Straus, et al. Virtual and augmented reality: potential applications in radiology. *Acta Radiologica*, 61(9):1258–1265, 2020.

[20] Alper H Duran, Munevver N Duran, Irfan Masood, Lynsey M Maciolek, and Huda Hussain. The additional diagnostic value of the three-dimensional volume rendering imaging in routine radiology practice. *Cureus*, 11(9), 2019.

[21] Paul S Calhoun, Brian S Kuszyk, David G Heath, Jennifer C Carley, and Elliot K Fishman. Three-dimensional volume rendering of spiral ct data: theory and method. *Radiographics*, 19(3):745–764, 1999.

[22] Andriy Fedorov, Reinhard Beichel, Jayashree Kalpathy-Cramer, Julien Finet, Jean-Christophe Fillion-Robin, Sonia Pujol, Christian Bauer, Dominique Jennings, Fiona Fennessy, Milan Sonka, et al. 3d slicer as an image computing platform for the quantitative imaging network. *Magnetic resonance imaging*, 30(9):1323–1341, 2012.

[23] Xiaolin Zhang, Kexin Zhang, Qiuling Pan, and Jincai Chang. Three-dimensional reconstruction of medical images based on 3d slicer. *Journal of Complexity in Health Sciences*, 2(1):1–12, 2019.

[24] Unity. https://docs.unity3d.com/2020.3/Documentation/Manual/.

[25] Pacchetti mrtk - mrtk2. https://learn.microsoft.com/it-it/windows/mixed-reality/mrtk-unity/mrtk2/packages/mrtk-packages?view=mrtkunity-2022-05, 2022.

[26] Hololens 2 - overview. https://www.microsoft.com/it-it/hololens/hardware.

[27] Agobiopsia polmonare tc guidata. https://www.gavazzeni.it/visite-ed-esami/agobiopsia-polmonare-tc-guidata/.

[28] Panoramica del rilevamento del codice a matrice. https://learn.microsoft.com/it-it/windows/mixed-reality/develop/advanced-concepts/qr-code-tracking-overview, 2022.