



**UNIVERSITÀ
DEGLI STUDI
DI PADOVA**



**DIPARTIMENTO
DI INGEGNERIA
DELL'INFORMAZIONE**

UNIVERSITÀ DEGLI STUDI PADOVA
DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

LAUREA TRIENNALE IN INGEGNERIA DELL'INFORMAZIONE

Tecniche di visualizzazione immersiva basate sul Deep Learning

LAUREANDA

Chiara Schiavo

Matricola 1216526

RELATORE

Prof. Simone Milani

Università di Padova

CORRELATORE

Dott.ssa Elena Camuffo

Università di Padova

ANNO ACCADEMICO 2021/2022
DATA DI LAUREA 25/11/2022

*Ai miei genitori
che hanno sempre creduto in me
e non mi hanno mai fatto mancare il loro supporto
a Gaia e Giulia
che rendono ogni giorno una commedia
ad Alessio
che sa riportarmi sempre all'equilibrio*

Sommario

Le applicazioni immersive e interattive hanno incontrato un notevole sviluppo nell'ultimo decennio. La loro crescente diffusione sta ponendo nuove sfide nel campo dell'elaborazione multimediale. Quando si realizza un'esperienza immersiva, gli aspetti più rilevanti da considerare sono la qualità dell'oggetto 3D rappresentato e la fluidità della visualizzazione nel caso l'utente si muova nell'ambiente. Inoltre è importante che la visualizzazione sia coerente con il posizionamento reciproco di utente e oggetto. La complessità del modello può essere adattata ai movimenti dell'utente e alla sua vicinanza all'oggetto, in modo da trovare un buon compromesso tra qualità dell'esperienza e carico computazionale. Questa tesi indaga il problema dell'ottimizzazione dei modelli 3D e le tecniche di visualizzazione basate su Deep Learning.

Indice

1	Tecnologie immersive	1
1.1	Concetto di immersività	1
1.2	Realtà Virtuale	3
1.3	Realtà Aumentata	3
1.4	Realtà Mista e Reality-Virtuality Continuum	4
1.5	Ambiti di applicazione	5
2	Ottimizzazione dei Modelli 3D	7
2.1	Rappresentazione dei modelli 3D	7
2.1.1	Mesh	8
2.1.2	Level Of Detail (LOD)	9
2.2	Problema dell'ottimizzazione	9
2.3	Approccio basato su Deep Learning	10
2.3.1	Cos'è il Deep Learning	10
2.3.2	Approccio Deep Learning al problema	10
3	Setup	11
3.1	Introduzione a Unity3D	11
3.2	Introduzione a Matlab	12
3.3	Dataset	13
3.4	Obiettivo dell'esperienza	15
3.5	Tipi di percorsi	15
3.5.1	Percorso 1	15
3.5.2	Percorso 2	16
3.5.3	Percorso 3	17
3.6	Acquisizione dei dati	17
3.7	Metriche di valutazione	18
3.7.1	SSIM	18
3.7.2	PSNR	18
4	Dati Sperimentali	19
4.1	Percorso 1	19
4.1.1	Variazione SSIM e PSNR lungo il percorso	19

INDICE

4.1.2	Elaborazione dei dati e calcolo dei valori medi	22
4.2	Percorso 2	23
4.2.1	Variazione SSIM e PSNR lungo il percorso	23
4.2.2	Elaborazione dei dati e calcolo dei valori medi	25
4.3	Percorso 3	26
4.3.1	Variazione SSIM e PSNR lungo il percorso	26
4.3.2	Elaborazione dei dati e calcolo dei valori medi	28
5	Conclusione	29

1

Tecnologie immersive

Negli ultimi decenni i contenuti multimediali hanno acquisito un ruolo sempre più predominante nella modellizzazione e trasmissione dell'informazione.

I sistemi immersivi di nuova generazione che non si limitano a dare maggiore verosimiglianza ai modelli 3D, ma consentono allo spettatore di immergersi in tali rappresentazioni e di interagire con esse stanno aprendo la possibilità di costruire realtà tridimensionali[1]. Queste innovazioni aprono le porte a nuove opportunità in numerosi settori, ma portano con sé anche molte sfide.

È necessario, infatti, che i contenuti multimediali siano di facile accesso e fruizione, che siano trasmissibili (spesso in tempo reale) senza interruzioni o ritardi e che la loro qualità non subisca degradazioni.

Trovare un buon compromesso tra questi aspetti è uno dei principali problemi da affrontare quando si ha a che fare con queste applicazioni.

La realtà virtuale e la realtà aumentata si pongono lo scopo di immergere l'utente in un'esperienza multisensoriale il più possibile realistica (a livello percettivo). Queste tecnologie si stanno diffondendo in ambiti applicativi sempre più ampi, uscendo dai confini del gaming e dell'intrattenimento, per trovare applicazione in campo industriale, medico-sanitario, nel settore della progettazione e dell'architettura, in ambito didattico e molto altro [13].

1.1 CONCETTO DI IMMERSIVITÀ

L'immersione in un ambiente virtuale è un'illusione mediata dalla tecnologia che, attraverso sistemi mimetici, genera stimoli capaci di coinvolgere i sensi dell'utente e di catturare la sua attenzione [2], "trasportandolo" in una realtà sintetica, ma percettivamente autentica. L'esperienza immersiva permette all'utente di sentire, percepire e interagire con elementi digitali. Questa esperienza può essere causata sia mentalmente che fisicamente:

1.1. CONCETTO DI IMMERSIVITÀ

Immersività mentale L'immersività mentale descrive uno stato in cui l'utente si sente profondamente impegnato, altamente coinvolto e pronto ad abbracciare la finzione. Per esempio, le persone lo sperimentano quando leggono un libro appassionante o guardano un film avvincente.

Immersività fisica L'immersività fisica si ottiene quando una persona riesce a sentirsi parte dell'azione, a muoversi e interagire con lo spazio. Un alto grado di immersività fisica si verifica quando vengono utilizzati dispositivi di input e output che impegnano effettivamente il maggior numero possibile di sensi dell'utente.

Presenza Nel contesto dell'immersività, il termine "presenza" è spesso usato per descrivere la sensazione soggettiva dell'utente di essere realmente nel mondo virtuale. A seconda del livello di dettaglio e di realismo dell'ambiente virtuale, lo spettatore lo percepisce come più o meno reale. Questa sensazione soggettiva può essere rafforzata da un alto grado di immersione fisica.

Mentre l'immersività mentale e la presenza sono considerati fenomeni psicologici, l'immersività fisica può essere intesa come un fenomeno tecnico o fisico, poiché i sensi sono impegnati attraverso stimoli fisici con l'aiuto delle tecnologie [13].

Con le diverse soluzioni immersive al giorno d'oggi disponibili, si riescono a raggiungere diversi gradi di immersione e di interazione tra utente ed elementi digitali. Nei sistemi di Realtà Virtuale (VR) l'utente è completamente immerso in un ambiente sintetico, che rimpiazza quello reale: la sensazione di immersione è totale e la percezione della realtà è completamente simulata. Nelle applicazioni di Realtà Aumentata (AR) elementi virtuali vengono sovrapposti al mondo reale. L'utente non viene quindi isolato dall'ambiente che lo circonda, ma lo vede integrato con contenuti virtuali. La Realtà Mista, è un insieme delle due tecnologie. Come l'AR, non sovrascrive completamente il mondo reale, ma permette all'utente lo stesso grado di interazione con oggetti virtuali e reali.



Figura 1.1: Confronto tra le diverse tecnologie immersive

1.2 REALTÀ VIRTUALE

La Realtà Virtuale, termine introdotto da Jaron Lanier nel 1989, è un ambiente esclusivamente digitale che simula la realtà effettiva creando un mondo parallelo. Questo mondo virtuale viene presentato ai nostri sensi mediante dispositivi informatici come visori, guanti e auricolari, che consentono una interazione in tempo reale con i diversi elementi sintetici. La VR è completamente immersiva e inganna i sensi inducendo l'utente a pensare di trovarsi in un ambiente diverso da quello reale [15].



Figura 1.2: Esempio di applicazione VR

1.3 REALTÀ AUMENTATA

Il termine Realtà Aumentata è stato introdotto nel 1990 da Tom Caudell. Deriva dal fatto che, secondo Caudell, questa tecnologia è progettata per "aumentare" il campo visivo dell'utente con informazioni (utili e che cambiano dinamicamente) necessarie nell'esecuzione di una certa attività corrente [3]. Nel 1997, Ronald T. Azuma, nel suo documento d'indagine [4], ne fornisce una definizione più accurata. Definisce la Realtà Aumentata come un sistema che:

- Combina oggetti reali e virtuali.
Nei sistemi di realtà aumentata, oggetti reali e virtuali coesistono allo stesso tempo e nello stesso luogo.
- Offre interazioni in tempo reale.
L'utente può interagire con i contenuti virtuali che rispondono alle sue azioni.
- Permette la registrazione di oggetti virtuali nel mondo fisico 3D.
C'è coerenza geometrica, prospettica e dimensionale tra gli oggetti virtuali e reali.

La realtà aumentata è una visione in tempo reale di contenuti digitali sovrapposte al mondo reale. Non rimpiazza il mondo fisico, come fa la realtà virtuale, ma lo implementa, espandendolo e arricchendolo con informazioni aggiuntive, che talvolta non possono essere rilevate e percepite direttamente.

1.4. REALTÀ MISTA E REALITY-VIRTUALITY CONTINUUM

La realtà aumentata migliora quindi la percezione e l'interazione dell'utente con il mondo reale. Le informazioni trasmesse dagli oggetti virtuali possono aiutarlo a svolgere attività del mondo reale. AR è un esempio specifico di ciò che Fred Brooks chiama Intelligence Amplification (IA): utilizzare il computer come strumento per rendere un compito più facile da eseguire per un essere umano [16].

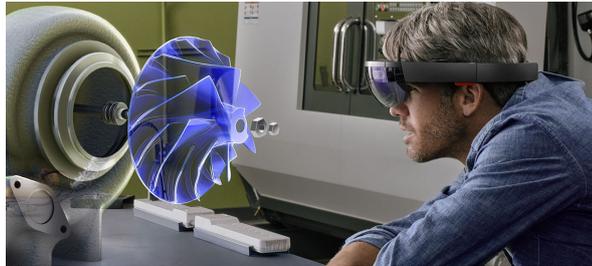


Figura 1.3: Esempio di applicazione AR

1.4 REALTÀ MISTA E REALITY-VIRTUALITY CONTINUUM

Con il termine Realtà Mista (MR) ci si riferisce all'insieme di tecnologie che combinano elementi reali con elementi virtuali. Questa classificazione segue il famoso Reality-Virtuality continuum teorizzato da Milgram e Kishino [6][8], che afferma l'esistenza di un intero spettro di tecnologie che va dall'ambiente puramente reale a quello interamente virtuale, passando per diversi gradi di "virtualizzazione". L'estremo sinistro definisce gli ambienti costituiti esclusivamente da oggetti reali. L'estremo destro definisce, invece, gli ambienti costituiti esclusivamente da oggetti virtuali. Il termine MR copre tutta la "zona grigia" al centro del continuum.

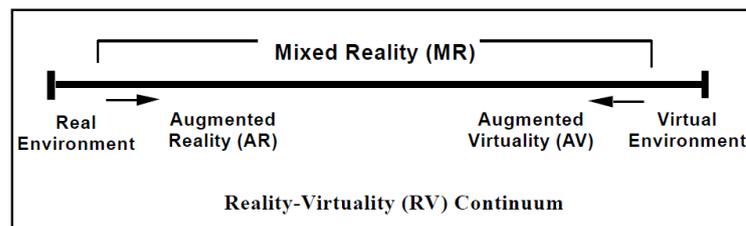


Figura 1.4: Reality-Virtuality Continuum

Oltre al Virtuality Continuum di Milgram e Kishino esistono poi altre definizioni di Realtà Mista. Una seconda definizione di MR è molto vicina a quella di realtà aumentata. Secondo questa definizione la MR sovrappone all'ambiente reale contenuti digitali e permette un livello di interazione con gli elementi virtuali pari a quello che si ha con gli oggetti reali, a differenza dell'AR che mostra delle informazioni o degli oggetti sopra il mondo reale, senza che si relazionino direttamente con esso.

1.5 AMBITI DI APPLICAZIONE

La VR ha avuto inizialmente un'accelerazione molto forte proveniente dai settori del gaming e dell'entertainment. L'utilizzo delle tecnologie immersive si è esteso negli anni ad altri ambiti, ed oggi i campi di applicazione sono molteplici, in seguito ne sono elencati alcuni.

Formazione AR e VR presentano un'estrema efficacia in applicazioni di addestramento e formazione. Meccanici, ingegneri, piloti e altri profili altamente specializzati possono apprendere i protocolli e le tecniche relative agli scenari in cui sono chiamati ad operare sfruttandone la ricostruzione virtuale. Con la capacità di ricreare esperienze del mondo reale con un alto livello di dettaglio, i programmi di formazione AR e VR sono in grado di coinvolgere gli utenti a situazioni realistiche e diverse, preparandoli per una qualsiasi delle variabili che potrebbero incontrare. Queste tecnologie possono essere inoltre utilizzate nella formazione medica: con la VR è possibile ricreare una sala operatoria in cui il personale medico può provare la sequenza prima di effettuare un intervento rischioso.

Trattamenti riabilitativi e salute mentale Le tecnologie immersive sono utilizzate nelle terapie riabilitative: AR e VR consentono di svolgere gli esercizi in un ambiente che riproduce le stesse identiche caratteristiche degli ambienti quotidiani, in modo da stimolare la multisensorialità del paziente, adattandoli man mano in modo dinamico rispetto alle abilità acquisite, monitorandone i progressi. AR e VR, inoltre, hanno un grande potenziale per trattare problemi di salute mentale come ansia e stress post-traumatico (PTSD), permettendo al paziente di affrontare le loro paure ed imparare a gestire i loro sintomi tramite la tecnica dell'esposizione.

Cultura e turismo Grazie alla VR è possibile visitare "virtualmente" luoghi di attrazione culturale senza dovervisi recare fisicamente. Le Gallerie degli Uffizi di Firenze e il Louvre di Parigi, per esempio, offrono visite virtuali di alcune esposizioni museali, mentre molte nazioni e regioni, come Irlanda e Trentino Alto Adige utilizzano esperienze VR per promuovere il patrimonio naturale e la cultura locale, con lo scopo di indurre gli utenti a viaggiare nei loro territori. VR permette inoltre di "passeggiare nel passato", grazie a ricostruzioni virtuali di siti storici, o di sperimentare luoghi difficilmente accessibili al pubblico. Anche la realtà aumentata permette di aggiungere valore alle visite culturali, grazie alla sovrapposizione di contenuti digitali a reperti e a monumenti, infatti permette all'utente di visualizzare dettagli storici e culturali relativi ai punti d'interesse durante il suo percorso, oppure, per esempio, di mostrare come si presentavano in un momento storico differente. L'AR può anche essere utilizzata per rendere interattive le esposizioni museali. Un esempio è offerto dal Museo Nazionale di Singapore: le 69 immagini della collezione

1.5. AMBITI DI APPLICAZIONE

di disegni di storia naturale che componevano un'installazione sono state trasformate in animazioni tridimensionali con cui i visitatori potevano interagire scaricando un'app.

Progettazione Le tecnologie immersive hanno un ruolo importante anche nella fase di progettazione e di test dei prodotti, permettendo risparmi di risorse, di tempo e diminuendo il rischio di errore nella costruzione. Grazie a VR e AR è possibile costruire un modello virtuale con le stesse caratteristiche dell'apparecchio fisico. In tal modo è possibile testarne il funzionamento e capire inoltre come il personale interagirà con esso, o come si inserirà nell'ambiente circostante.



Figura 1.5: Ambiti applicativi di AR e VR

2

Ottimizzazione dei Modelli 3D

La crescente diffusione di applicazioni di Realtà Virtuale e Realtà Aumentata su dispositivi mobili e indossabili pone costantemente nuove sfide nel processo di elaborazione e fruizione delle informazioni multimediali. [17] L'utilizzo quotidiano di queste tecnologie impone agli sviluppatori di creare i modelli utilizzabili da dispositivi dalle prestazioni differenti senza compromettere la qualità dell'esperienza immersiva. Quando si ha a che fare con queste applicazioni, è importante che i sistemi garantiscano immersività attraverso:

- Rendering in tempo reale di oggetti 3D;
- Risoluzione ad alta fedeltà degli ambienti;
- Interazione fluida con il mondo virtuale;

Per creare un'esperienza immersiva è quindi necessario rendere la visualizzazione dei modelli 3D naturale e continua ottimizzando il carico computazionale richiesto al dispositivo. La complessità del modello può essere adattata ai movimenti dell'utente e alla sua vicinanza all'oggetto.

2.1 RAPPRESENTAZIONE DEI MODELLI 3D

Nelle applicazioni immersive gli elementi ed ambienti virtuali devono essere rappresentati in modo consistente all'esperienza che si vuole simulare. La rappresentazione tridimensionale rispetta i requisiti di realismo, profondità e senso di presenza di cui necessitano queste applicazioni. I modelli 3D sono un'approssimazione della superficie e della forma dell'oggetto reale visualizzato. Ci sono tre principali tecniche di rappresentazione di modelli 3D:

- **Mesh:** la superficie dell'oggetto è approssimata da una maglia poligonale; più sono fitti i poligoni, e dunque maggiore è il numero di

2.1. RAPPRESENTAZIONE DEI MODELLI 3D

vertici che li compongono, migliore è l'approssimazione dell'oggetto originale;

- **Point Cloud:** il modello tridimensionale viene generato a partire da una fitta serie di punti localizzati in uno spazio di riferimento (x,y,z) . I punti, oltre che dalle informazioni di posizione, possono essere caratterizzati dall'intensità;
- **Voxel:** questa rappresentazione, a differenza delle precedenti, è basata sul volume. L'oggetto infatti, viene rappresentato attraverso piccole unità volumetriche dette voxel.

I modelli mesh, attualmente, dominano la computer grafica 3D, quindi ci concentreremo prevalentemente su questa rappresentazione.

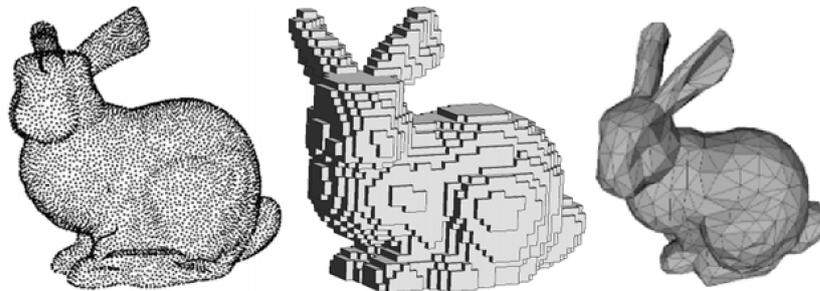


Figura 2.1: Il modello 3D 'Stanford Bunny' rappresentato con tre diverse tecniche: Point Cloud (sinistra), Voxel (centro), Mesh triangolare (destra)

2.1.1 MESH

La rappresentazione tridimensionale mediante mesh prevede che le superfici dell'oggetto siano descritte utilizzando un certo numero di punti nello spazio 3D. Questi punti sono chiamati vertici e sono interconnessi da archi che definiscono delle superfici poligonali costituenti la mesh. È preferibile utilizzare poligoni semplici, per questa ragione considereremo le maglie costituite da triangoli. La decimazione della mesh triangolare è uno strumento importante per trovare un buon compromesso tra qualità dell'esperienza e complessità computazionale del modello. L'obiettivo della decimazione della mesh è di ridurre il numero totale di triangoli mantenendo una buona approssimazione della geometria originale. Questo permette di rimuovere i dettagli che l'occhio non è in grado di distinguere alleggerendo il carico computazionale. Un'eccessiva degradazione del modello porta a distorsioni geometriche che compromettono la qualità della visione e di conseguenza dell'esperienza.

2.1.2 LEVEL OF DETAIL (LOD)

Il LOD o Livello di Dettaglio L è un indice di valutazione di un modello Mesh 3D correlato al numero di triangoli utilizzati nella rappresentazione. La decimazione dei triangoli che costituiscono il modello determina differenti livelli LOD. Il LOD di un modello nelle applicazioni AR/VR influenza fortemente l'efficienza del sistema e la qualità percepita dall'utente. Adottare un grande numero di triangoli risulta non necessario qualora il punto di vista sia distante, dal momento che, i dettagli, non possono essere apprezzati dall'utente a causa della dimensione ridotta del modello renderizzato. Al contrario un LOD basso restituisce una visualizzazione di bassa qualità se l'utente si trova vicino all'oggetto [17].

L_0 rappresenta il LOD del modello originale, ossia quello con il maggior numero di triangoli (modello di maggiore qualità).

I livelli di dettaglio L_i vengono generati a partire dal modello originale riducendo il numero di vertici, $i = 1, 2, 3$ indica il livello di decimazione dei triangoli della mesh.

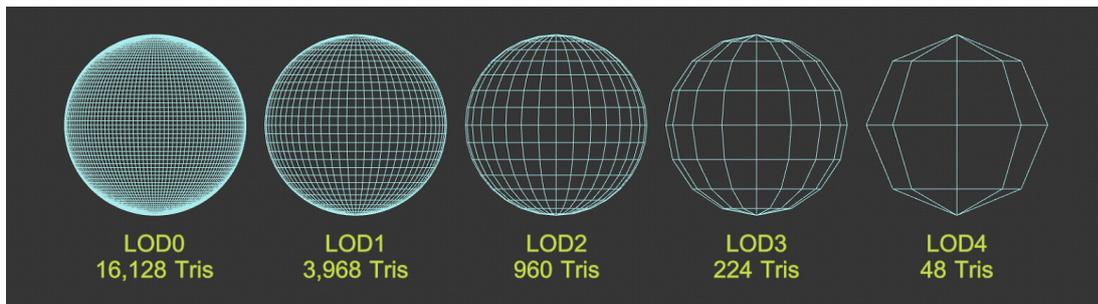


Figura 2.2: Rappresentazione di una sfera utilizzando diversi LOD

2.2 PROBLEMA DELL'OTTIMIZZAZIONE

La qualità dell'esperienza è determinata dalla qualità dei modelli e dalla fluidità della rappresentazione. La qualità del modello è influenzata dal LOD, dalla posizione e dall'orientamento dell'utente rispetto alla posizione dell'oggetto virtuale, mentre la frequenza di aggiornamento FPS, che determina la fluidità, è correlata al carico computazionale della CPU o della scheda grafica, e influenzata dalle caratteristiche del modello 3D visualizzato. Un modello con alto livello di dettaglio risulta sicuramente realistico, ma grava maggiormente sulla GPU compromettendo la fluidità di rappresentazione [17]. Un modello fortemente semplificato, invece, è facilmente renderizzabile, ma potrebbe influire negativamente sulla qualità dell'esperienza. È dunque necessario trovare un buon compromesso tra qualità e complessità dei modelli. A tal fine, è possibile adattare il numero di poligoni nei modelli 3D in base alla loro distanza dallo spettatore o dalla telecamera e all'interazione con l'ambiente virtuale degli oggetti 3D. In un ambiente MR, in cui l'utente è in grado di muoversi

2.3. APPROCCIO BASATO SU DEEP LEARNING

liberamente intorno all'oggetto, la posizione, la prospettiva e la velocità dell'utente rispetto all'oggetto determinano i dettagli che è in grado di distinguere. Ciò suggerisce che è necessario utilizzare più LOD dello stesso oggetto 3D da cui scegliere quello ottimale sulla base delle considerazioni precedenti. In questo modo il carico di lavoro della CPU o della scheda grafica viene ridotto e la qualità percepita dall'utente non cala. Tuttavia è difficile stimare il numero di triangoli che ottimizzerebbero sia il LOD del modello 3D che la frequenza di fotogrammi, in quanto i fattori da tenere in considerazione sono molteplici.

2.3 APPROCCIO BASATO SU DEEP LEARNING

2.3.1 COS'È IL DEEP LEARNING

Il Deep Learning è una sottocategoria del Machine Learning (che letteralmente viene tradotto come apprendimento automatico) e indica quella branca dell'intelligenza artificiale che fa riferimento agli algoritmi ispirati alla struttura e alla funzione del cervello, chiamati reti neurali artificiali.

Da un punto di vista scientifico, si può dire che il Deep Learning sia l'apprendimento da parte delle macchine attraverso dati appresi grazie all'utilizzo di algoritmi.

2.3.2 APPROCCIO DEEP LEARNING AL PROBLEMA

Un approccio basato su Deep Learning al problema sopra presentato viene proposto in [17]. Questa strategia mira a ottimizzare la visualizzazione di oggetti 3D selezionando in modo adattivo il set di parametri più adatto. L'approccio si basa sulla previsione di alcune metriche (spiegate nel capitolo 3).

3

Setup

3.1 INTRODUZIONE A UNITY3D

Unity3D è un motore grafico multiplatforma sviluppato da Unity Technologies che consente lo sviluppo di videogiochi e altri contenuti interattivi, quali visualizzazioni architettoniche o animazioni 3D in tempo reale. Di seguito sono elencati i componenti principali di Unity3D [18].

Asset Un Asset è un qualsiasi supporto o dato che può essere utilizzato in un progetto; può provenire da un file creato da un altro applicativo, come ad esempio un modello 3D, un file audio o un'immagine. È possibile creare alcuni semplici tipi di Asset direttamente in Unity, ad esempio un mixer audio o una texture di rendering.

GameObjects e Components I GameObjects sono gli elementi fondamentali in Unity. Ogni oggetto nel progetto è un GameObject; una telecamera, per definire il punto di vista da cui renderizzare la scena è, ad esempio, un GameObject. Essi fungono da contenitori per i Components, che ne implementano le funzionalità. I Components contengono infatti le proprietà, che è possibile modificare, per definire il comportamento di un GameObject. Un GameObject ha sempre una Component Transform collegata, che ne determina la posizione, la rotazione e la scala. Gli altri Components possono essere aggiunti dall'apposito menu dell'editor o da uno script. È possibile visualizzare un elenco dei Components collegati ad un GameObject nella finestra Inspector, dopo aver selezionato il GameObject.

Scripts Lo scripting è un ingrediente essenziale in tutte le applicazioni realizzate in Unity. Gli scripts possono essere utilizzati per creare effetti grafici, controllare il comportamento fisico degli oggetti o implementare un sistema di intelligenza artificiale personalizzato per i personaggi del gioco. I linguaggi di programmazione supportati

3.2. INTRODUZIONE A MATLAB

sono javascript e C#. I metodi di base sono Start(), che include le azioni eseguite all'avvio dell'applicazione, e Update(), che viene richiamato ad ogni frame.



Figura 3.1: Interfaccia di Unity

3.2 INTRODUZIONE A MATLAB

MATLAB (abbreviazione di Matrix Laboratory) è una piattaforma di programmazione e calcolo numerico utilizzata per l'analisi di dati, lo sviluppo di algoritmi e la creazione di modelli [19]. Consente di manipolare matrici, visualizzare funzioni e dati, creare interfacce utente, e integrarsi con altri programmi. MATLAB è usato da milioni di persone nell'industria e nelle università per via dei suoi numerosi strumenti a supporto dei più disparati campi di studio applicativi. L'interfaccia principale di MATLAB è composta da diverse finestre, le principali sono quattro:

- **Command window:** è una finestra dell'interfaccia principale di MATLAB nella quale è possibile digitare comandi supportati e visualizzare a schermo in tempo reale i risultati;
- **Workspace:** è lo spazio di lavoro (o spazio di memoria) contenente le variabili dichiarate;
- **Current directory:** è una finestra che permette di esplorare il contenuto delle cartelle sul proprio supporto di memoria;
- **Command history:** è una finestra in cui sono elencati tutti i comandi digitati di recente, divisi per ora e data;

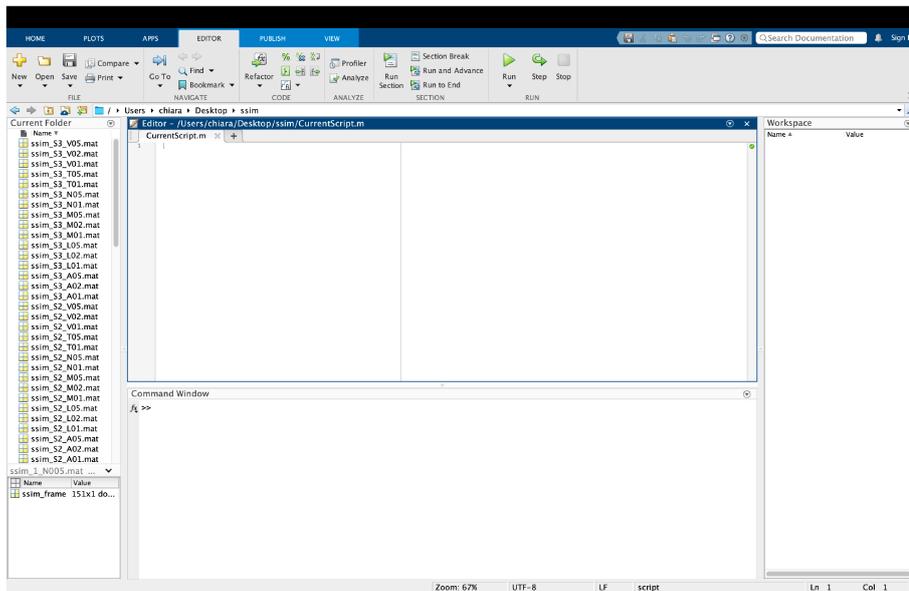


Figura 3.2: Interfaccia Matlab

3.3 DATASET

Per i test utilizziamo sette differenti modelli tridimensionali, ognuno presente a diversi livelli di decimazione della mesh, ossia a differenti LOD. Tutti i modelli sono considerati privi di attributi (ossia senza texture, colori, etc) al fine limitare la valutazione alla complessità geometrica. I modelli sono stati importati in Unity e dimensionati in modo da essere contenuti in un cubo $1m \times 1m \times 1m$ ($1m$ corrisponde circa ad una unità in Unity) centrato nell'origine del sistema di riferimento cartesiano. Di seguito, sono riportati i modelli.

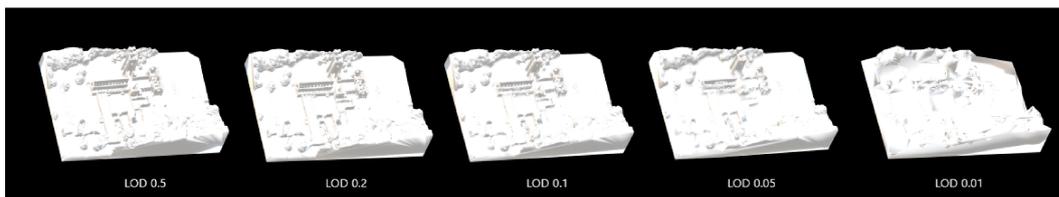


Figura 3.3: Modello "Lackock Abbey" ai diversi LOD utilizzati

3.3. DATASET



Figura 3.4: Modello "Lidded Ewer" ai diversi LOD utilizzati



Figura 3.5: Modello "Mercedes-Benz" ai diversi LOD utilizzati

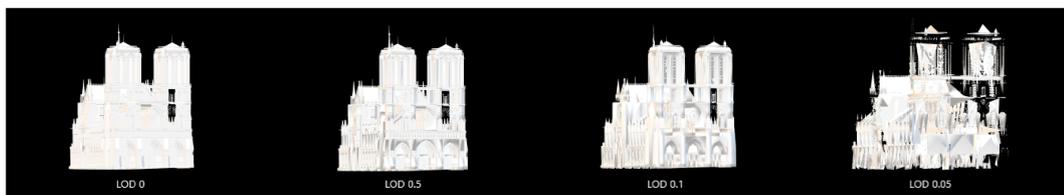


Figura 3.6: Modello "Notre Dame" ai diversi LOD utilizzati



Figura 3.7: Modello "Owl" ai diversi LOD utilizzati

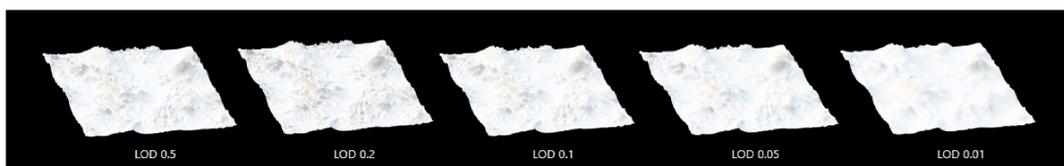


Figura 3.8: Modello "Death Valley" ai diversi LOD utilizzati

3.4 OBIETTIVO DELL'ESPERIENZA

Lo studio effettuato consiste nella valutazione dei vari modelli, da un punto di vista oggettivo, in un ipotetico scenario di AR. Sono stati scelti 3 diversi percorsi (figura 3.9) atti a rappresentare lo spostamento di un utente intorno ad un modello 3D. Lungo ciascuno di questi percorsi, sono stati raccolti dati (i.e. FPS, numero di vertici del modello visibili dal punto di vista della camera, screenshots) per ogni modello, per ogni LOD. Lo scopo dell'esperienza è valutare come variano i valori oggettivi (FVC, FPS) e quelli di qualità (PSNR,SSIM) al variare dell'LOD e della distanza tra utente e oggetto.

3.5 TIPI DI PERCORSI

È stato scelto, invece di studiare la visualizzazione dei modelli in AR, di valutare sequenze di immagini acquisite dalla telecamera in Unity3D e il derivante video lungo tre percorsi prestabiliti che possono rappresentare gli spostamenti di un utente intorno ad un modello 3D in AR, per esempio nel corso di un'esperienza immersiva museale. Si ipotizza che l'utente esegua uno spostamento continuo senza bruschi cambi di direzione e che mantenga lo sguardo rivolto verso il modello. Questa soluzione, utilizzata in [20], ci permette di ottenere dati oggettivi e di più facile elaborazione.

Sono stati quindi stabiliti tre percorsi da far effettuare alla telecamera attorno al modello. Si è scelto di mantenere la velocità della telecamera pressoché costante, in modo da poter fare considerazioni sulla qualità dei modelli percepita in funzione della distanza e orientazione camera-modello.

I tre percorsi sono stati definiti in differenti Script che sono stati poi assegnati come *Component* alla telecamera in Unity. In particolare, mediante script, sono state modificate le proprietà *Transform.position* e *Transform.rotation* della telecamera, delineando il suo spostamento. In tutti i percorsi la telecamera si muove sul piano xz , la componente y del vettore di posizione rimane quindi a 0. Si assume, inoltre, che il punto di vista della telecamera sia sempre indirizzato verso il modello in esame.

3.5.1 PERCORSO 1

La telecamera si avvicina all'oggetto in linea retta lungo l'asse x , poi ruota intorno al modello mantenendo una distanza costante dall'oggetto.

In coordinate polari, quindi in funzione del raggio ρ e dell'angolo θ , definito come angolo di rotazione intorno all'asse y , avremo che, nel tratto rettilineo di percorso l'angolo θ rimane costante e varia ρ da 5 a 0.8, nel secondo tratto invece la telecamera ruota attorno al modello con raggio costante $\rho = 0.8$, mentre l'angolo θ varia da -90° a 0° . Il percorso termina dopo una rotazione di 90° in senso orario attorno al modello.

3.5. TIPI DI PERCORSI

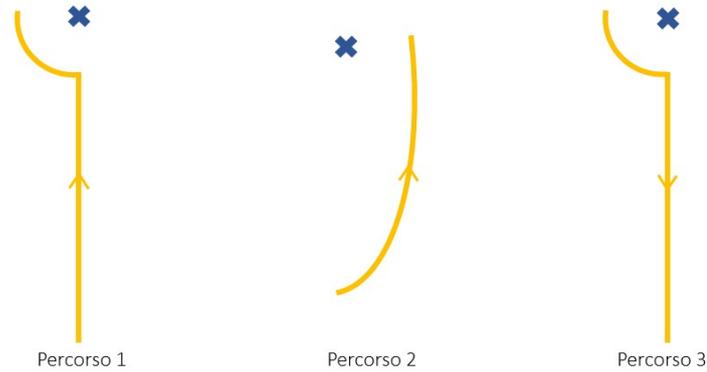


Figura 3.9: In giallo sono rappresentati i tre percorsi che effettua la telecamera (la freccia indica il verso di percorrenza), le croci blu indicano la posizione del modello



Figura 3.10: Percorso 1 effettuato dalla telecamera in Unity

3.5.2 PERCORSO 2

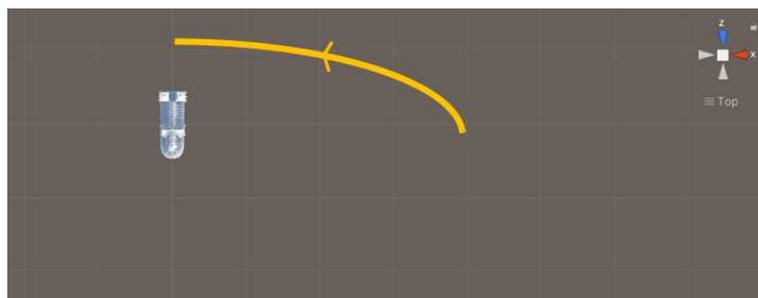


Figura 3.11: Percorso 2 effettuato dalla telecamera in Unity

Il percorso descrive un arco di circonferenza. La circonferenza scelta è centrata in $(0, 0, -7)$ e interseca l'asse delle ascisse in $x = -4$, in questo modo la distanza di partenza e la vicinanza massima della telecamera al modello non differiscono particolarmente dalle rispettive distanze nei percorsi 1 e 3. Quindi come nel percorso precedente la telecamera si avvicina all'oggetto, ma in questo caso, ρ e θ variano simultaneamente. Attorno all'asse y la telecamera spazza un angolo $\theta = 90^\circ$. Questo percorso è il più breve tra i tre in esame.

3.5.3 PERCORSO 3



Figura 3.12: Percorso 3 effettuato dalla telecamera in Unity

Il percorso 3 è equivalente al percorso 1, ma con direzione di percorrenza opposta. La telecamera parte dunque in prossimità dell'oggetto ad una distanza $\rho = 0.8$ ed esegue una rotazione di un quarto di circonferenza attorno all'oggetto con raggio costante. Arrivata sull'asse x , la camera si allontana progressivamente fino a raggiungere una distanza dall'origine $\rho = 5$.

3.6 ACQUISIZIONE DEI DATI

I tre percorsi sono stati utilizzati per guardare e valutare la variazione di qualità tra i modelli presentati nel paragrafo 3.3 originali e le loro versioni con differenti LOD. La telecamera di Unity esegue in successione i tre percorsi per ogni versione dei modelli, generando nel corso dell'esecuzione di ogni percorso un file di testo contenente le posizioni occupate dalla telecamera frame per frame, la sua orientazione e il numero di *Frame Vertex Count (FVC)*, un indicatore che conteggia il numero di vertici della mesh visibili da una data posizione.

Il programma è stato poi eseguito nuovamente per generare una serie di screenshots ("fotografie dal punto di vista della camera"). L'idea è che il programma esegua uno screenshot per ogni posizione salvata nel file precedentemente acquisito.

I set di dati ottenuti eseguendo il percorso attorno a modelli con LOD_0 vengono considerati come riferimento al fine del calcolo della variazione della qualità. Le sequenze di immagini acquisite lungo un certo percorso, utilizzando un determinato modello con mesh decimata, vengono confrontate con le acquisizioni di riferimento, al fine di valutare come varia la differenza di qualità tra modello in esame e modello di riferimento lungo il percorso.

Le sequenze di screenshots sono state inoltre montate in brevi video, per poter visualizzare ciò che l'utente vedrebbe eseguendo il percorso.

3.7 METRICHE DI VALUTAZIONE

Per valutare la qualità dei modelli si utilizzano le due seguenti metriche, che come anticipato nel capitolo 2, sono anche alla base all'approccio Deep Learning al problema dell'ottimizzazione proposto in [17].

3.7.1 SSIM

L'indice di somiglianza strutturale (SSIM) è una metrica percettiva che quantifica il degrado della qualità dell'immagine causato da un'elaborazione come la compressione dei dati o, nel nostro caso, dalla decimazione della mesh. SSIM è un indice di riferimento: la misurazione o la previsione della qualità dell'immagine si basa su un'immagine iniziale non compressa o priva di distorsioni come riferimento.

È possibile calcolare SSIM in MATLAB, mediante la funzione $ssim(Im,Ref)$ che richiede come parametri l'immagine da confrontare e la relativa immagine di riferimento.

3.7.2 PSNR

Il termine *Peak Signal-to-Noise Ratio* (PSNR) è un'espressione che identifica il rapporto tra il valore massimo possibile (potenza) di un segnale e la potenza del rumore di distorsione che influisce sulla qualità della sua rappresentazione. Poiché molti segnali hanno una gamma dinamica molto ampia (rapporto tra i valori più grandi e più piccoli possibili di una quantità variabile) il PSNR è solitamente espresso in termini di scala dei decibel logaritmica.

È possibile calcolare PSNR in MATLAB, mediante la funzione $psnr(Im,Ref)$ che richiede come parametri l'immagine da confrontare e la relativa immagine di riferimento.

4

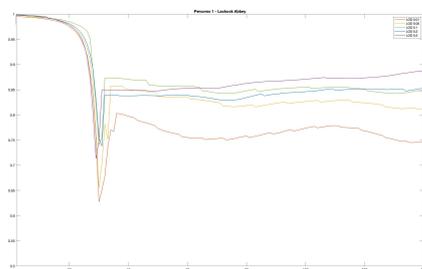
Dati Sperimentali

4.1 PERCORSO 1

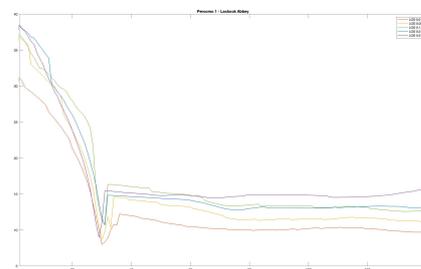
4.1.1 VARIAZIONE SSIM E PSNR LUNGO IL PERCORSO

I grafici che seguono mostrano come variano SSIM e PSNR puntualmente lungo il percorso. Ogni grafico riporta i valori ottenuti percorrendo il primo percorso attorno ad uno dei sette modelli. Sono stati utilizzati dei diversi colori per distinguere le acquisizioni ottenute con LOD differenti.

Si può notare che i modelli con mesh più sottocampionata sono meno performanti. La differenza di prestazione rispetto ai modelli con LOD maggiore, cresce con il diminuire della lontananza dal modello. I grafici mostrano un decremento rapido dell'SSIM e del PSNR nel primo tratto, dovuto all'avvicinarsi della telecamera al modello. Nel punto di raccordo tra traslazione e rotazione della camera i dati sono particolarmente affetti da rumore, nell'ultimo tratto invece i valori rimangono circa costanti, difatti la telecamera gira attorno al modello con raggio costante.



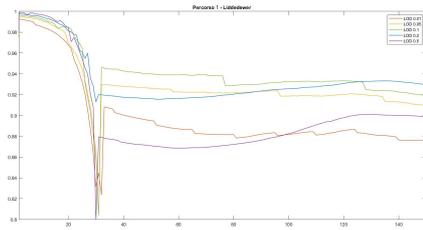
(a) SSIM



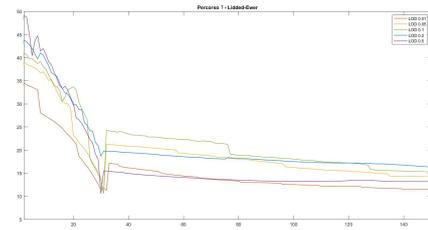
(b) PSNR

Figura 4.1: Lackock Abbey

4.1. PERCORSO 1

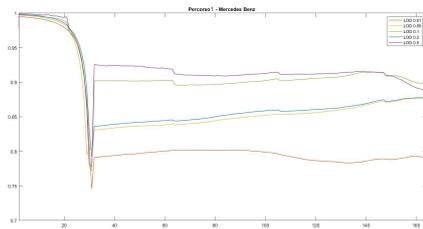


(a) SSIM

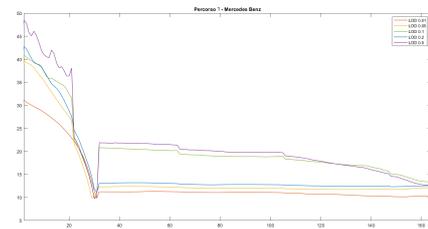


(b) PSNR

Figura 4.2: Litted-Ewer

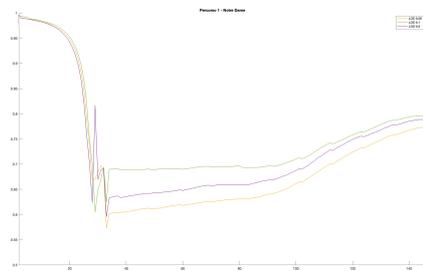


(a) SSIM

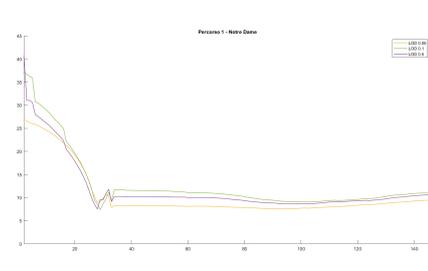


(b) PSNR

Figura 4.3: Mercedes-Benz

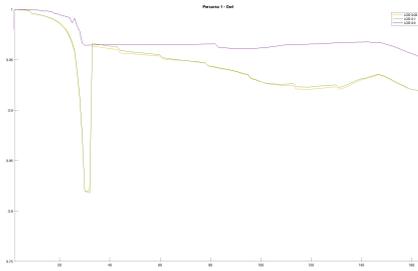


(a) SSIM

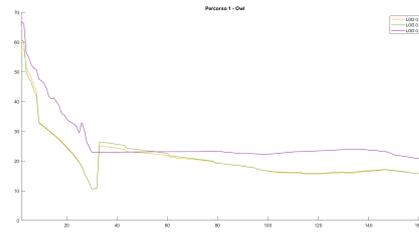


(b) PSNR

Figura 4.4: Notre Dame

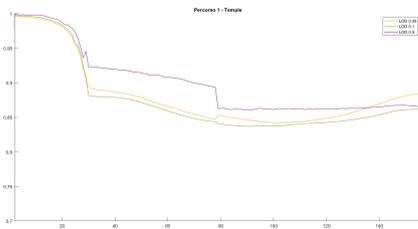


(a) SSIM

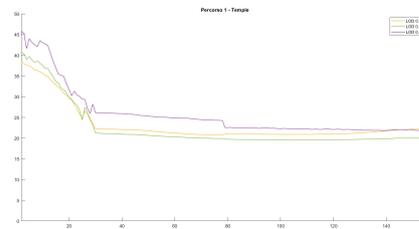


(b) PSNR

Figura 4.5: Owl

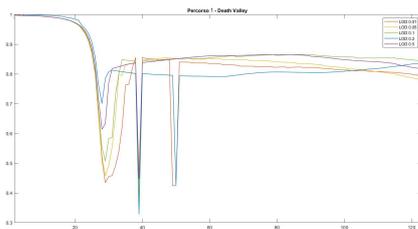


(a) SSIM

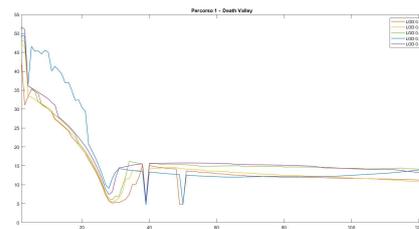


(b) PSNR

Figura 4.6: Temple



(a) SSIM



(b) PSNR

Figura 4.7: Death Valley

4.1. PERCORSO 1

4.1.2 ELABORAZIONE DEI DATI E CALCOLO DEI VALORI MEDI

Nelle seguenti tabelle sono riportati i valori medi di SSIM e PSNR lungo il percorso 1 per ogni livello LOD di ogni modello. I dati trovati sono coerenti con gli LOD, infatti nella tabella si può notare gli a LOD superiori corrispondono SSIM e PSNR maggiori.

	LOD 0.5	LOD 0.2	LOD 0.1	LOD 0.05	LOD 0.01
Lackock Abbey	0.87637	0.86125	0.87058	0.84683	0.79711
Lidded-Ewer	0.89451	0.92853	0.93359	0.92327	0.89301
Mercedes-Benz	0.91753	0.87181	0.90907	0.86652	0.82015
Notre Dame	0.73927	/	0.75473	0.71851	/
Owl	0.9636	/	0.93787	0.93714	/
Temple	0.89493	/	0.87054	0.87762	/
Valley	0.85884	0.82671	0.85536	0.83662	0.81578

Tabella 4.1: SSIM mediati lungo il percorso per i differenti modelli a differenti LODs

	LOD 0.5	LOD 0.2	LOD 0.1	LOD 0.05	LOD 0.01
Lackock Abbey	17.1549	16.476	16.8218	14.9522	12.8717
Lidded-Ewer	17.4014	20.763	21.4491	19.4628	15.2368
Mercedes-Benz	21.2807	15.8248	20.2337	14.7948	13.2142
Notre Dame	12.0678	/	12.8096	10.6414	/
Owl	25.9509	/	20.7306	20.4926	/
Temple	25.5662	/	22.239	23.1309	/
Valley	17.1816	17.0379	16.6277	14.9024	14.2525

Tabella 4.2: PSNR mediati lungo il percorso per i differenti modelli a differenti LODs

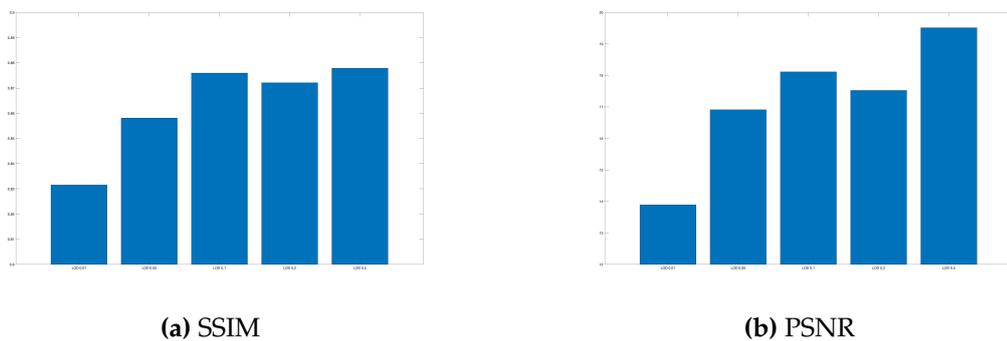


Figura 4.8: Media di tutti i dati acquisiti ai differenti LOD lungo il percorso 1

4.2 PERCORSO 2

4.2.1 VARIAZIONE SSIM E PSNR LUNGO IL PERCORSO

I seguenti grafici mostrano come variano SSIM e PSNR puntualmente lungo il percorso. Ogni grafico riporta i valori ottenuti percorrendo il secondo percorso attorno ad uno dei sette modelli. Come nel paragrafo precedente, diversi colori sono utilizzati per distinguere le acquisizioni ottenute con LOD differenti.

Anche in questo caso si può notare che i modelli con mesh maggiormente decimata sono meno performanti. I grafici mostrano una diminuzione di SSIM e PSNR procedendo lungo il percorso. La telecamera, infatti, percorrendo un arco si avvicina al modello (vedi figura 3.9 in centro), non sono presenti punti critici.

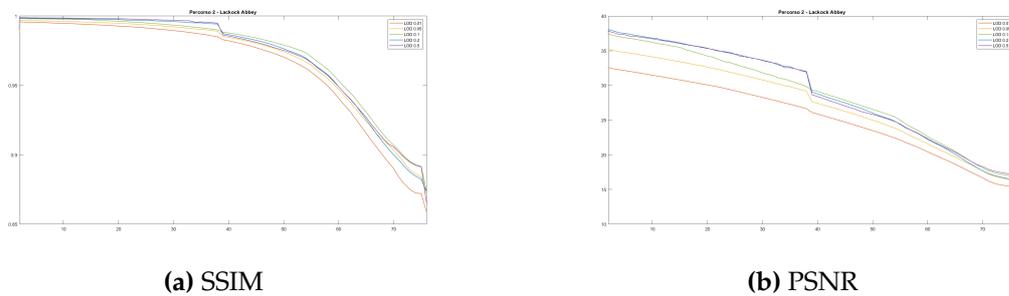


Figura 4.9: Lackock Abbey

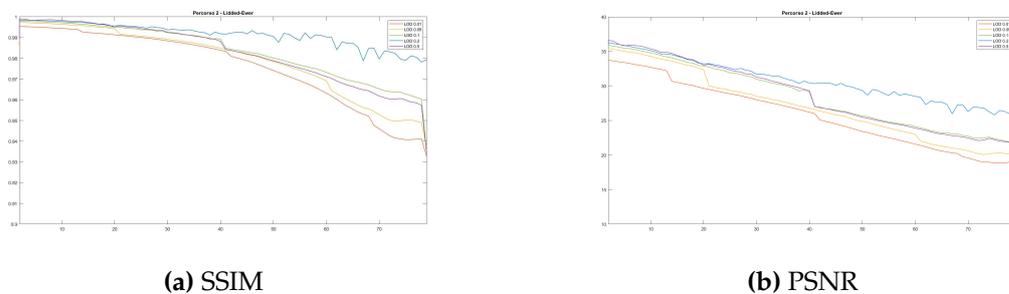


Figura 4.10: Lidded-Ewer

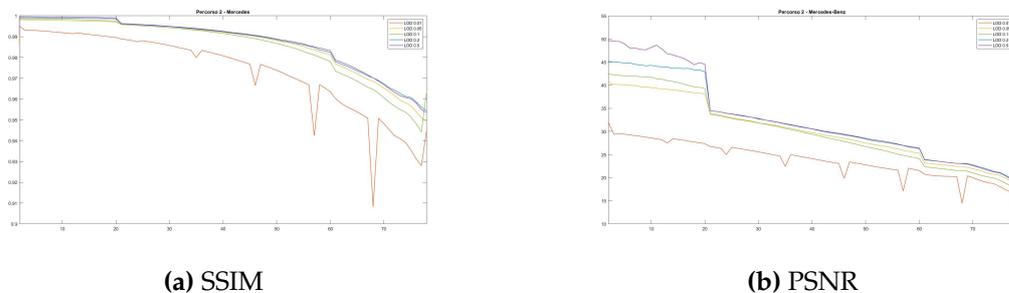
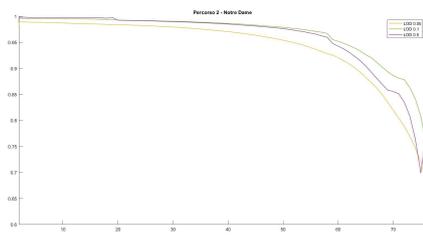
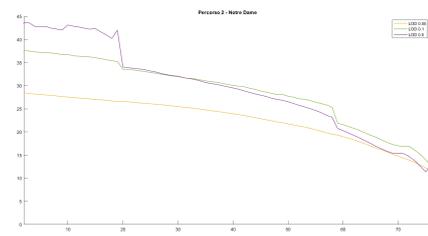


Figura 4.11: Mercedes-Benz

4.2. PERCORSO 2

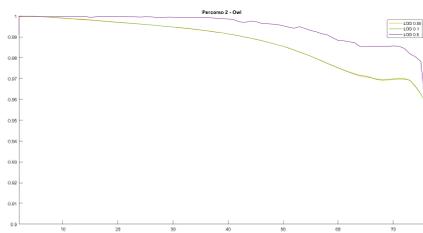


(a) SSIM

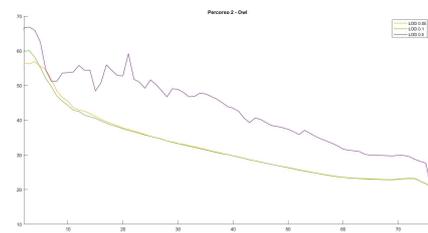


(b) PSNR

Figura 4.12: Notre Dame

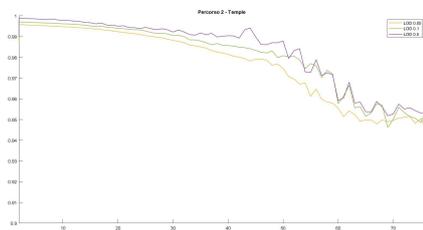


(a) SSIM

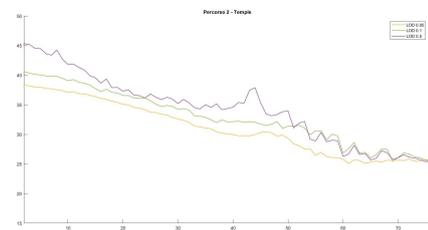


(b) PSNR

Figura 4.13: Owl

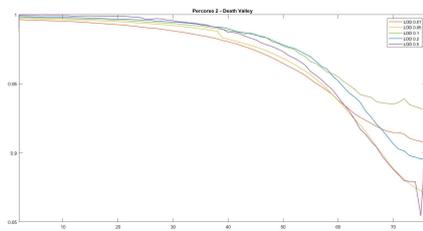


(a) SSIM

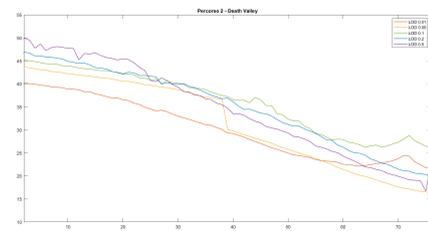


(b) PSNR

Figura 4.14: Temple



(a) SSIM



(b) PSNR

Figura 4.15: Death Valley

4.2.2 ELABORAZIONE DEI DATI E CALCOLO DEI VALORI MEDI

Nelle seguenti tabelle sono riportati i valori medi di SSIM e PSNR lungo il percorso 2 per ogni livello LOD di ogni modello. Anche in questo caso i valori in tabella confermano ciò che ci si aspettava, a LOD maggiori corrispondono infatti SSIM e PSNR più elevati.

	LOD 0.5	LOD 0.2	LOD 0.1	LOD 0.05	LOD 0.01
Lackock Abbey	0.95934	0.95872	0.95997	0.95671	0.95223
Lidded-Ewer	0.97015	0.97888	0.97088	0.96677	0.96355
Mercedes-Benz	0.9751	0.97489	0.97267	0.97378	0.9611
Notre Dame	0.94579	/	0.95333	0.92846	/
Owl	0.98161	/	0.97446	0.97444	/
Temple	0.96981	/	0.96763	0.96361	/
Valley	0.95613	0.9608	0.9648	0.95294	0.95518

Tabella 4.3: SSIM mediati lungo il percorso per i differenti modelli a differenti LODs

	LOD 0.5	LOD 0.2	LOD 0.1	LOD 0.05	LOD 0.01
Lackock Abbey	28.5642	28.5878	28.2207	26.8825	25.0094
Lidded-Ewer	28.1703	30.365	28.0498	26.6374	25.437
Mercedes-Benz	33.2954	31.5052	29.7901	29.7647	23.6028
Notre Dame	29.0067	/	28.3451	22.4346	/
Owl	42.6075	/	31.9497	32.0256	/
Temple	33.585	/	32.2959	30.477	/
Valley	34.0879	34.4918	35.559	31.0647	29.7181

Tabella 4.4: PSNR mediati lungo il percorso per i differenti modelli a differenti LODs

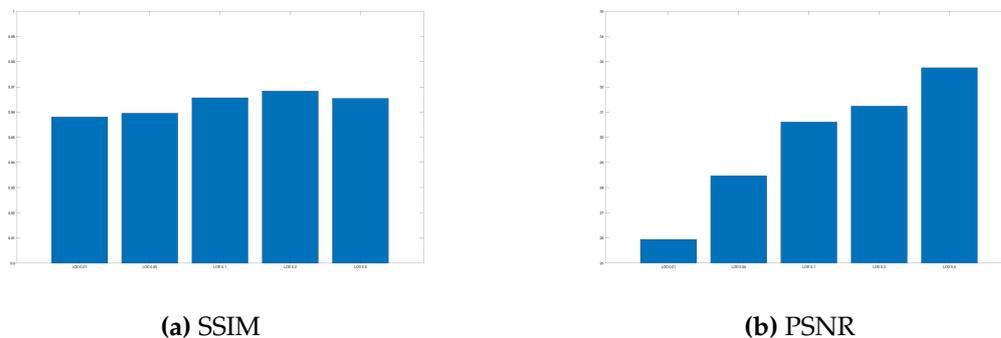


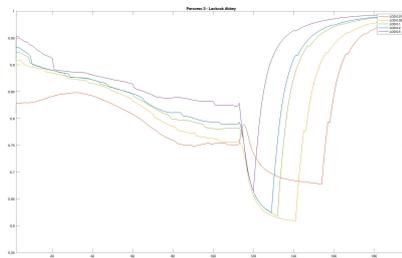
Figura 4.16: Media di tutti i dati acquisiti ai differenti LOD lungo il percorso 2

4.3 PERCORSO 3

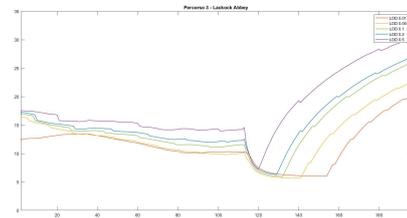
4.3.1 VARIAZIONE SSIM E PSNR LUNGO IL PERCORSO

I seguenti grafici mostrano come variano SSIM e PSNR puntualmente lungo il percorso. Ogni grafico riporta i valori ottenuti percorrendo il terzo percorso attorno ad uno dei sette modelli. Come nei paragrafi precedenti, diversi colori sono utilizzati per distinguere le acquisizioni ottenute con LOD differenti.

Dai grafici si può notare un primo tratto in cui i valori in esame rimangono pressochè stabili ed un secondo tratto in cui, invece, SSIM e PSNR crescono velocemente a causa dell'allontanamento progressivo della telecamera dal modello. Come in "Percorso 1" nel punto di giunzione tra rotazione e traslazione della telecamera i dati sono rumorosi.

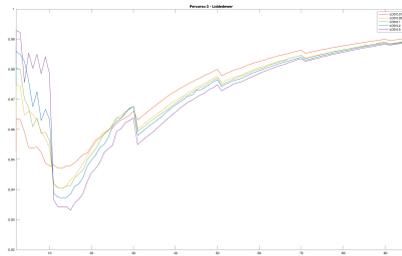


(a) SSIM

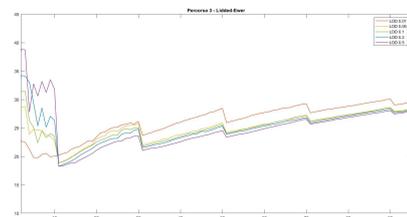


(b) PSNR

Figura 4.17: Lackock Abbey

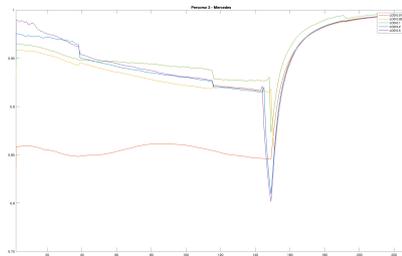


(a) SSIM

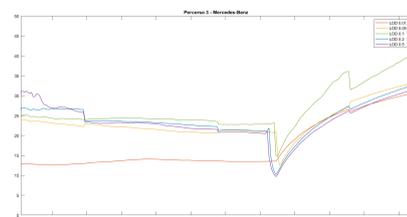


(b) PSNR

Figura 4.18: Lidded-Ewer

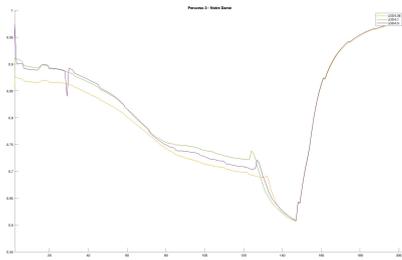


(a) SSIM

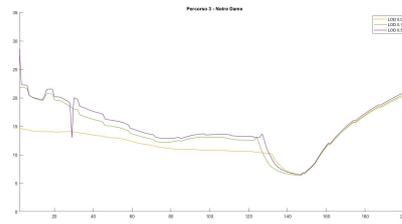


(b) PSNR

Figura 4.19: Mercedes-Benz

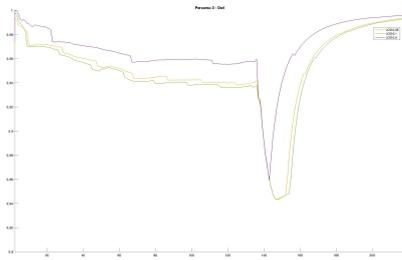


(a) SSIM

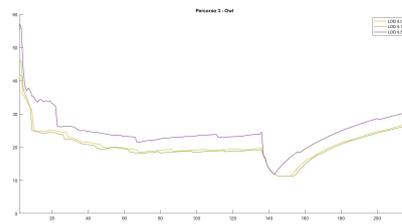


(b) PSNR

Figura 4.20: Notre Dame

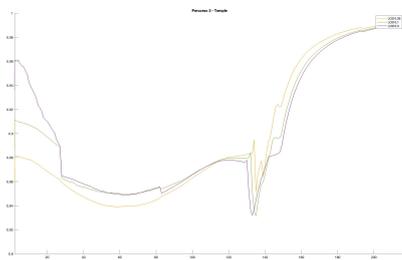


(a) SSIM

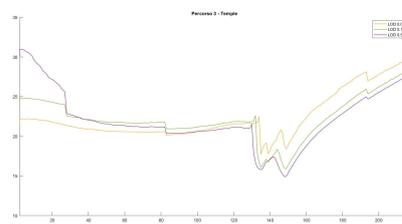


(b) PSNR

Figura 4.21: Owl

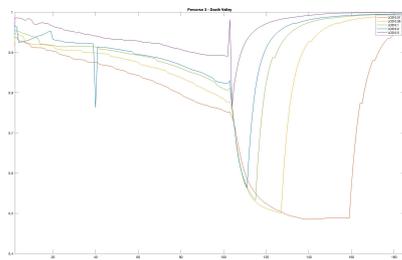


(a) SSIM

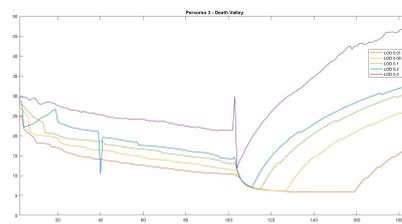


(b) PSNR

Figura 4.22: Temple



(a) SSIM



(b) PSNR

Figura 4.23: Death Valley

4.3. PERCORSO 3

4.3.2 ELABORAZIONE DEI DATI E CALCOLO DEI VALORI MEDI

Nelle seguenti tabelle sono riportati i valori medi di SSIM e PSNR lungo il percorso 3 per ogni livello LOD di ogni modello. Come nei casi precedenti, i dati raccolti sono coerenti con gli LOD.

	LOD 0.5	LOD 0.2	LOD 0.1	LOD 0.05	LOD 0.01
Lackock Abbey	0.8948	0.86315	0.85472	0.82675	0.80391
Lidded-Ewer	0.96137	0.96206	0.96219	0.9624	0.96327
Mercedes-Benz	0.94791	0.95376	0.94298	0.92909	0.89063
Notre Dame	0.81121	/	0.81308	0.80017	/
Owl	0.96391	/	0.94484	0.94778	/
Temple	0.89979	/	0.90564	0.90548	/
Valley	0.94863	0.90675	0.89205	0.8492	0.74691

Tabella 4.5: SSIM mediati lungo il percorso per i differenti modelli a differenti LODs

	LOD 0.5	LOD 0.2	LOD 0.1	LOD 0.05	LOD 0.01
Lackock Abbey	17.8616	15.2116	14.3644	12.4998	11.2997
Lidded-Ewer	24.7249	24.7927	24.761	24.8073	25.8554
Mercedes-Benz	23.4186	25.3506	23.0336	20.1018	17.0397
Notre Dame	14.8693	/	14.2317	12.5422	/
Owl	24.6687	/	20.236	20.638	/
Temple	22.0787	/	23.2286	24.6545	/
Valley	29.1701	20.6155	18.766	15.6693	11.342

Tabella 4.6: PSNR mediati lungo il percorso per i differenti modelli a differenti LODs

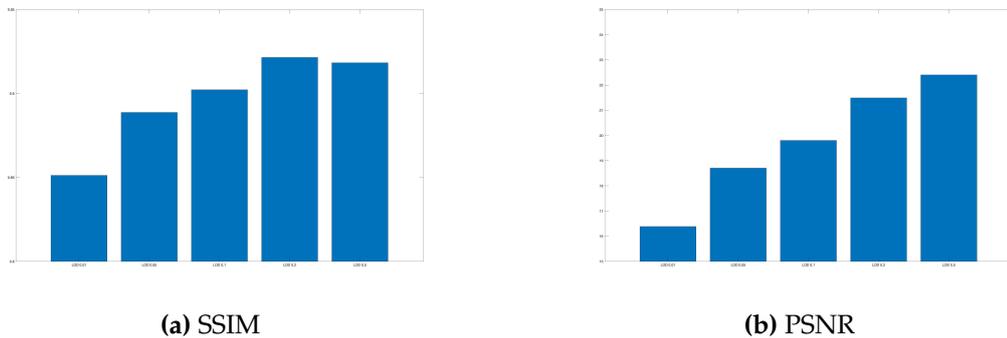


Figura 4.24: Media di tutti i dati acquisiti ai differenti LOD lungo il percorso 3

5

Conclusione

In questa tesi è stato studiato il problema dell'ottimizzazione dei modelli 3D. Il problema dell'ottimizzazione del livello di dettaglio relativo a diversi modelli 3D, è stato studiato, per creare un'esperienza immersiva rendendo la visualizzazione dei modelli fluida e naturale. Se l'utente è in prossimità dell'oggetto ne riesce a cogliere anche i dettagli più piccoli e dunque, affinché l'esperienza risulti realistica, il modello deve essere di alta qualità; se l'oggetto osservato invece è distante dall'utente, risulta essere più piccolo e i dettagli non sono distinguibili, dunque un modello di qualità inferiore risulta comunque ottimale alla vista dell'utente.

Lo studio effettuato consiste nella valutazione di sette modelli, da un punto di vista oggettivo, in un ipotetico scenario di AR. Sono stati scelti 3 percorsi atti a rappresentare lo spostamento di un utente intorno ad un modello 3D. Ciascun percorso è stato eseguito attorno a ciascun modello nella sua versione originale e nelle varianti con LOD inferiore. Durante l'esperienza sono stati acquisiti dati e immagini, per poter poi fare un'analisi oggettiva della qualità percepita.

I risultati ottenuti sono in linea con quanto ci si aspetta: SSIM e PSNR calano al diminuire della distanza tra camera e modello. Al diminuire della distanza, inoltre, la variazione di qualità tra modello originale e relativo modello con LOD inferiore, aumenta.

In studi futuri, ci si propone di aumentare il dataset e di aggiungere all'analisi ulteriori metriche di valutazione basate, per esempio, sul confronto della qualità dei video prodotti. Inoltre si prevede di automatizzare la procedura di ottimizzazione mediante tecniche di Deep Learning.

Bibliografia

- [1] <https://www.treccani.it/enciclopedia/comunicazione-e-nuove-tecnologie>
- [2] Chenyan Zhang, "The Why, What, and How of Immersive Experience", Department of Electronic Systems, Norwegian University of Science and Technology (NTNU), 7491 Trondheim, Norway
- [3] Thomas P. Caudell, David Mizell, "Augmented reality: An application of heads-up display technology to manual manufacturing processes", Conference Paper - February 1992
- [4] Ronald T. Azuma, "A Survey of Augmented Reality", Teleoperators and Virtual Environments, August 1997
- [5] Mandeep Handa et al. "Immersive Technology - Uses, Challenges and Opportunities"
- [6] Paul Milgram et al. "Augmented Reality: A class of displays on the reality-virtuality continuum"
- [7] Jon Peddie et al. "Augmented Reality - Where we all live", Springer
- [8] Richard Skarbez et al. "Revisiting Milgram and Kishino's Reality-Virtuality Continuum"
- [9] Kundalakesi M. et al. "A Study of Virtual Reality"
- [10] Alexandra Kitson et al. "Immersive Interactive Technologies for Positive Change: A Scoping Review and Design Considerations", 2018
- [11] A. Pavithra et al. "An Emerging Immersive Technology-A Survey"
- [12] Mark Weiser, "The Computer for the 21st Century"
- [13] <https://makerfairerome.eu/it/tecnologie-immersive-i-progetti-a-tema-vr-e-ar-a-maker-faire-rome-2022/>
- [14] <https://www.cad-schroer.it/novita/articoli/immersione-e-mondi-immersivi-la-sfida-del-presente/>
- [15] <https://www.intel.it/content/www/it/it/tech-tips-and-tricks/virtual-reality-vs-augmented-reality.html>
- [16] Brooks P F. The Computer Scientist as Toolsmith II. CACM, 1996: 39(3), p. 61-68
- [17] Elena Camuffo et al. "Deep 3D Model Optimization for Immersive and Interactive Applications"
- [18] <https://docs.unity3d.com/Manual/Glossary.html>

