



**Politecnico
di Torino**

Politecnico di Torino

Corso di Laurea Magistrale in Ingegneria Informatica

A.a. 2025/2026

Sessione di laurea Marzo 2026

Analisi comparativa di strumenti per il Gaussian Splatting e valutazione in ambienti immersivi

Relatori:

Andrea Bottino

Francesco Strada

Candidata:

Ernesta Maria Sichetti

Sommario

Negli ultimi anni il Gaussian Splatting si è affermato come tecnica per la ricostruzione di ambienti tridimensionali a partire da acquisizioni di ambienti reali.

Questa tesi analizza e confronta vari strumenti per il Gaussian Splatting, partendo da un'analisi qualitativa, per poi concentrarsi in modo più specifico su due tipologie di strumenti fondamentali: i trainer open source per la generazione delle ricostruzioni e i visualizzatori per la loro esplorazione in modalità immersiva. Su questi strumenti sono state condotte valutazioni tecniche sia visive che quantitative.

Sulla base dei risultati ottenuti è stata selezionata una pipeline operativa composta da un trainer open source e da un visualizzatore per la fruizione delle ricostruzioni in realtà virtuale. Gli ambienti tridimensionali generati sono stati infine utilizzati in uno studio sperimentale con utenti per analizzare la percezione dell'immersività e la qualità dell'esperienza in scenari ricostruiti tramite Gaussian Splatting visualizzati in realtà virtuale.

Il lavoro fornisce quindi un quadro comparativo degli strumenti disponibili per il Gaussian Splatting e indicazioni utili per la scelta di soluzioni adatte allo sviluppo di ambienti immersivi.

Indice

Elenco delle tabelle	VI
Elenco delle figure	VIII
1 Introduzione	1
1.1 Evoluzione della Computer Grafica e Rappresentazione 3D	1
1.2 Dalla Structure from Motion al Neural Rendering	2
1.3 Neural Radiance Fields (NeRF)	3
1.4 3D Gaussian Splatting	4
1.5 Obiettivi della tesi e panoramica dei risultati	4
1.6 Struttura della tesi	6
2 Stato dell'arte	7
2.1 Structure-from-Motion e calibrazione delle camere	7
2.1.1 Structure-from-Motion	7
2.1.2 COLMAP	8
2.2 3D Gaussian Splatting	9
2.2.1 Panoramica del metodo	9
2.2.2 Rappresentazione della scena tramite gaussiane anisotrope .	10
2.2.3 Ottimizzazione e controllo adattivo della densità	11
2.2.4 Rendering differenziabile	12
2.2.5 Implementazione di riferimento	13
2.3 Ecosistema degli strumenti per 3D Gaussian Splatting	13
2.4 Trainer per 3D Gaussian Splatting	13
2.4.1 Trainer commerciali	14
2.4.2 Trainer open source	15
2.5 Visualizzatori, editor e integrazioni per 3D Gaussian Splatting . . .	17
2.5.1 Visualizzatori	17
2.5.2 Strumenti di editing	18
2.5.3 Integrazioni con game engine	18
2.5.4 Visualizzatori XR	19

2.6	Sintesi degli strumenti analizzati	19
2.7	Selezione degli strumenti per l'analisi comparativa	21
3	Analisi comparativa	23
3.1	Metodologia	23
3.1.1	Pipeline di analisi tecnica	23
3.1.2	Condizioni sperimentali	26
3.1.3	Hardware	26
3.1.4	Metriche di valutazione	26
3.2	Overview degli strumenti	27
3.2.1	Trainer	28
3.2.2	Visualizzatori, editor e strumenti di integrazione	32
3.3	Analisi tecnica delle pipeline di training open source	34
3.4	Analisi tecnica scena indoor	35
3.4.1	Benchmark quantitativo	36
3.4.2	Analisi visiva delle ricostruzioni raw	38
3.4.3	Impatto delle operazioni di pulizia	40
3.4.4	Analisi visiva delle ricostruzioni dopo il post-processing	42
3.4.5	Sintesi dei risultati sulla scena indoor	45
3.5	Analisi tecnica scena outdoor	45
3.5.1	Benchmark quantitativo	46
3.5.2	Analisi visiva delle ricostruzioni raw	47
3.5.3	Impatto delle operazioni di pulizia	50
3.5.4	Analisi visiva delle ricostruzioni dopo il post-processing	52
3.5.5	Sintesi dei risultati sulla scena outdoor	55
3.6	Confronto tra scena indoor e scena outdoor	55
3.6.1	Differenze quantitative nelle ricostruzioni raw	55
3.6.2	Differenze strutturali tra ricostruzioni indoor e outdoor	57
3.6.3	Differenze nelle operazioni di pulizia	57
3.6.4	Impatto quantitativo delle operazioni di pulizia	57
3.6.5	Sintesi del confronto tra scenari	59
3.7	Selezione dello scenario per l'analisi dei visualizzatori XR	59
3.8	Analisi tecnica dei visualizzatori XR	60
3.8.1	Analisi visiva delle ricostruzioni nei visualizzatori XR	61
3.8.2	Confronto tra i visualizzatori XR	63
3.8.3	Scelta del visualizzatore per l'analisi delle prestazioni	63
3.8.4	Analisi delle prestazioni dei modelli nel visualizzatore XR	64
3.8.5	Sintesi dei risultati dell'analisi dei visualizzatori XR	66
3.9	Selezione degli strumenti per ambienti immersivi	67
4	Ambiente immersivo e sperimentazione sugli utenti	68

4.1	Descrizione generale dell'esperimento	68
4.1.1	Ambiente immersivo	68
4.1.2	Condizioni sperimentali	69
4.1.3	Struttura dell'esperimento	71
4.2	Flusso di lavoro dello studio sperimentale	71
4.3	Partecipanti	72
4.4	Ipotesi di ricerca	73
4.5	Questionari e strumenti di valutazione	74
4.6	Analisi dei dati	75
4.7	Risultati	76
4.7.1	Quality of Experience	77
4.7.2	Qualità visiva percepita	78
4.7.3	Comfort dell'esperienza immersiva	78
4.7.4	Prestazioni hardware	79
4.7.5	Preferenze degli utenti	80
4.8	Discussione dei risultati	80
4.9	Limitazioni dello studio	82
5	Conclusioni e sviluppi futuri	83
5.1	Sintesi del percorso progettuale	83
5.2	Risultati sperimentali e analisi dei risultati	84
5.3	Contributi del lavoro	84
5.4	Sviluppi futuri	85
5.5	Considerazioni finali	86
	Bibliografia	87

Elenco delle tabelle

2.1	Panoramica degli strumenti per 3D Gaussian Splatting analizzati nello stato dell'arte	20
3.1	Informazioni generali sui trainer commerciali.	28
3.2	Caratteristiche delle pipeline di training dei trainer commerciali. . .	29
3.3	Informazioni generali sui trainer open source.	30
3.4	Caratteristiche delle pipeline di training dei trainer open source. . .	31
3.5	Informazioni generali sui visualizzatori, editor e strumenti di integrazione.	32
3.6	Ruolo e caratteristiche degli strumenti di visualizzazione e integrazione.	33
3.7	Metriche originali delle pipeline di training sulla scena indoor. . . .	36
3.8	Metriche normalizzate rispetto al numero di gaussiane generate. . .	36
3.9	Impatto delle operazioni di pulizia sul numero di gaussiane della scena indoor.	41
3.10	Impatto delle operazioni di pulizia sulla dimensione dei modelli della scena indoor.	41
3.11	Metriche originali delle pipeline di training sulla scena outdoor. . .	46
3.12	Metriche normalizzate rispetto al numero di gaussiane generate. . .	46
3.13	Impatto delle operazioni di pulizia sul numero di gaussiane della scena outdoor.	51
3.14	Impatto delle operazioni di pulizia sulla dimensione dei modelli della scena outdoor.	51
3.15	Confronto tra scena indoor e outdoor in termini di numero di gaussiane e dimensione dei modelli.	55
3.16	Confronto tra scena indoor e outdoor in termini di tempo di training.	56
3.17	Riduzione percentuale del numero di gaussiane dopo il post-processing.	58
3.18	Riduzione percentuale della dimensione dei modelli dopo il post-processing.	58
3.19	Valutazione qualitativa delle ricostruzioni nel visualizzatore Unity-VR-Gaussian-Splatting (ninjamode).	61

3.20	Valutazione qualitativa delle ricostruzioni nel visualizzatore GaussianSplattingVRViewerUnity (clarte53).	62
3.21	Prestazioni di rendering delle ricostruzioni nel visualizzatore Unity-VR-Gaussian-Splatting (ninjamode).	64
4.1	Distribuzione demografica dei partecipanti coinvolti nello studio. . .	73
4.2	Metriche utilizzate per l'analisi dei dati dell'esperimento.	76
4.3	Metriche di Quality of Experience osservate nelle due condizioni sperimentali.	77
4.4	Riepilogo della conferma delle ipotesi di ricerca.	81

Elenco delle figure

1.1	Workflow generale del lavoro di tesi: dall'analisi degli strumenti Gaussian Splatting alla selezione della pipeline operativa e alla valutazione sperimentale in ambiente immersivo.	6
2.1	Pipeline incrementale di Structure-from-Motion. Il processo include l'estrazione delle feature, la ricerca delle corrispondenze tra immagini, la verifica geometrica e la ricostruzione incrementale della scena tramite registrazione delle immagini, triangolazione dei punti e <i>bundle adjustment</i> . Figura tratta da [3].	8
2.2	Pipeline del metodo 3D Gaussian Splatting. A partire da una nuvola di punti ottenuta tramite Structure-from-Motion (SfM) viene inizializzato un insieme di gaussiane tridimensionali. Durante il <i>training</i> i parametri delle gaussiane vengono ottimizzati e la loro densità adattata dinamicamente tramite adaptive density control . Il rendering delle viste avviene tramite un rasterizzatore differenziabile progettato per supportare il training e la visualizzazione in tempo reale. Figura tratta da [6].	10
2.3	Schema del processo di adaptive density control nel 3D Gaussian Splatting. In presenza di sotto-ricostruzione della geometria una gaussiana può essere duplicata (clone), mentre in caso di over-reconstruction una gaussiana può essere suddivisa in più primitive (split). Figura tratta da [6].	12
3.1	Pipeline sperimentale utilizzata per l'analisi comparativa degli strumenti Gaussian Splatting e la selezione della pipeline operativa da utilizzare in ambiente immersivo.	25
3.2	Confronto delle principali metriche quantitative delle pipeline di training sulla scena indoor: numero di gaussiane generate, dimensione dei modelli esportati e tempo di training.	37
3.3	Ricostruzione raw della scena indoor generata da Inria gaussian-splatting.	38

3.4	Ricostruzione raw della scena indoor generata da gsplat.	39
3.5	Ricostruzione raw della scena indoor generata da OpenSplat.	39
3.6	Ricostruzione raw della scena indoor generata da Nerfstudio.	40
3.7	Ricostruzione raw della scena indoor generata da LichtFeld Studio.	40
3.8	Riduzione percentuale del numero di gaussiane e della dimensione dei modelli dopo le operazioni di pulizia nella scena indoor.	42
3.9	Ricostruzione della scena indoor dopo le operazioni di pulizia (Inria gaussian-splatting).	43
3.10	Ricostruzione della scena indoor dopo le operazioni di pulizia (gsplat).	43
3.11	Ricostruzione della scena indoor dopo le operazioni di pulizia (OpenSplat).	44
3.12	Ricostruzione della scena indoor dopo le operazioni di pulizia (Nerfstudio).	44
3.13	Ricostruzione della scena indoor dopo le operazioni di pulizia (LichtFeld Studio).	45
3.14	Confronto delle principali metriche quantitative delle pipeline di training sulla scena outdoor: numero di gaussiane generate, dimensione dei modelli esportati e tempo di training.	47
3.15	Ricostruzione raw della scena outdoor generata da Inria gaussian-splatting.	48
3.16	Ricostruzione raw della scena outdoor generata da gsplat.	48
3.17	Ricostruzione raw della scena outdoor generata da OpenSplat.	49
3.18	Ricostruzione raw della scena outdoor generata da Nerfstudio.	49
3.19	Ricostruzione raw della scena outdoor generata da LichtFeld Studio.	50
3.20	Riduzione percentuale del numero di gaussiane e della dimensione dei modelli dopo le operazioni di pulizia nella scena outdoor.	52
3.21	Ricostruzione della scena outdoor dopo le operazioni di pulizia (Inria gaussian-splatting).	52
3.22	Ricostruzione della scena outdoor dopo le operazioni di pulizia (gsplat).	53
3.23	Ricostruzione della scena outdoor dopo le operazioni di pulizia (OpenSplat).	53
3.24	Ricostruzione della scena outdoor dopo le operazioni di pulizia (Nerfstudio).	54
3.25	Ricostruzione della scena outdoor dopo le operazioni di pulizia (LichtFeld Studio).	54
3.26	Confronto tra scena indoor e outdoor nelle principali metriche di training: numero di gaussiane generate e tempo di training per le pipeline analizzate.	56
3.27	Riduzione percentuale del numero di gaussiane dopo il post-processing nelle scene indoor e outdoor.	58

3.28	Frame rate medio delle ricostruzioni nel visualizzatore Unity-VR-Gaussian-Splatting (ninjamode).	65
3.29	Tempo medio di rendering dei frame e tempo massimo osservato durante l'esecuzione nel visualizzatore Unity-VR-Gaussian-Splatting (ninjamode).	65
4.1	Lobby iniziale dell'esperimento VR utilizzata per la fase di familiarizzazione dei partecipanti.	69
4.2	Ricostruzione della scena indoor utilizzata nella condizione sperimentale di confronto con il video della scena reale.	70
4.3	Ricostruzione dell'ambiente reale utilizzato nella condizione sperimentale di confronto diretto con la stanza fisica.	70
4.4	Workflow dello studio sperimentale: dalla definizione delle ipotesi alla verifica dei risultati ottenuti dall'esperimento in ambiente immersivo.	72
4.5	Distribuzione del titolo di studio dei partecipanti.	73
4.6	Esempio di domanda del questionario visualizzata direttamente all'interno dell'ambiente immersivo.	75
4.7	Confronto tra realismo percepito e senso di presenza nelle due condizioni sperimentali.	77
4.8	Livello medio di cybersickness nelle due condizioni sperimentali.	79
4.9	Preferenza degli utenti per la condizione percepita come più immersiva.	80

Capitolo 1

Introduzione

1.1 Evoluzione della Computer Grafica e Rappresentazione 3D

La computer grafica tridimensionale ha l'obiettivo di rappresentare e visualizzare ambienti tridimensionali digitali. Storicamente, la rappresentazione delle scene nella computer grafica tridimensionale si è basata su **modelli geometrici espliciti**, nei quali la geometria della scena è descritta direttamente tramite primitive geometriche. Tra le primitive più utilizzate vi sono i **poligoni**, che permettono di costruire strutture geometriche più complesse come le **mesh poligonali**.

Le mesh rappresentano le superfici degli oggetti attraverso vertici, spigoli e facce e costituiscono da decenni lo standard de facto nella computer grafica. In particolare, le **mesh triangolari** sono ampiamente utilizzate grazie alla loro efficienza computazionale e alla perfetta integrazione con le pipeline hardware delle GPU [1].

Nel corso dell'evoluzione della computer grafica sono state sviluppate diverse strategie alternative per rappresentare la geometria tridimensionale di una scena. Tra le più diffuse vi sono le **nuvole di punti** (*point clouds*), che descrivono una scena tramite un insieme discreto di punti nello spazio tridimensionale. Anche in questo caso la geometria della scena è rappresentata in modo esplicito, poiché la posizione dei punti definisce direttamente la struttura tridimensionale dell'ambiente. Le nuvole di punti sono frequentemente utilizzate nelle pipeline di ricostruzione tridimensionale a partire da immagini o da sensori di profondità [2].

Un'altra famiglia di rappresentazioni è costituita dalle **griglie voxel** (*voxel grids*), nelle quali lo spazio tridimensionale viene discretizzato in una griglia regolare di celle volumetriche. Ogni cella, detta **voxel**, può memorizzare informazioni relative alla densità volumetrica, al colore o ad altre proprietà della scena [2].

Parallelamente allo sviluppo delle mesh poligonali, la modellazione geometrica ha

fatto largo uso di primitive parametriche esplicite, tra cui curve di Bézier, B-spline e superfici NURBS. Queste rappresentazioni consentono di descrivere superfici lisce e continue attraverso formulazioni matematiche parametriche e sono ampiamente utilizzate in contesti come la modellazione CAD e la progettazione industriale [1].

Accanto alle rappresentazioni esplicite, nella computer grafica e nella ricostruzione tridimensionale sono state sviluppate anche **rappresentazioni implicite**, nelle quali la geometria della scena non è descritta direttamente tramite primitive geometriche, ma viene definita come l'insieme dei punti dello spazio che soddisfano una determinata funzione matematica

Una volta definita la rappresentazione geometrica della scena, il passo successivo consiste nel generare l'immagine finale tramite un processo di rendering. Nelle applicazioni di grafica interattiva il metodo di visualizzazione predominante è storicamente la **rasterizzazione**. Questo processo consiste nel proiettare le primitive geometriche sul piano immagine e nel determinare quali pixel siano coperti da ciascun triangolo della scena. Durante questa fase vengono inoltre risolti i problemi di visibilità e profondità tramite algoritmi dedicati, tra cui il noto metodo dello *z-buffering*, che consente di determinare quale superficie sia visibile per ogni pixel dell'immagine [1].

Sebbene la rasterizzazione sia estremamente efficiente, essa presenta limiti intrinseci nel simulare fenomeni ottici complessi come riflessioni multiple, rifrazioni o effetti di illuminazione globale. Per ottenere una rappresentazione più accurata del comportamento della luce sono stati sviluppati metodi alternativi basati sulla simulazione fisica del trasporto della luce. Tra questi, il **ray tracing** rappresenta uno degli approcci più noti: invece di proiettare direttamente la geometria sul piano immagine, il ray tracing simula il percorso dei raggi luminosi nello spazio tridimensionale, calcolando le interazioni tra luce e superfici per determinare il colore finale di ciascun pixel [1].

Storicamente, il ray tracing è stato considerato troppo costoso dal punto di vista computazionale per applicazioni interattive, poiché richiede il calcolo di numerose intersezioni tra raggi e superfici della scena. Negli ultimi anni, tuttavia, i progressi nell'hardware grafico e negli algoritmi di accelerazione del ray tracing hanno progressivamente ridotto questo divario prestazionale, rendendo possibile l'utilizzo di tecniche basate sui raggi anche in contesti interattivi.

1.2 Dalla Structure from Motion al Neural Rendering

Con lo sviluppo delle tecniche di acquisizione digitale, la ricerca nel campo della computer vision e della computer grafica si è progressivamente orientata verso metodi di ricostruzione automatica della geometria tridimensionale a partire da

immagini reali. In questo contesto, una delle tecniche più diffuse è la **Structure from Motion** (SfM), che consente di stimare simultaneamente la struttura tridimensionale di una scena e le pose delle camere a partire da un insieme di immagini acquisite da punti di vista differenti [3].

Le pipeline di SfM prevedono generalmente una fase iniziale di estrazione delle *feature* e ricerca delle corrispondenze tra immagini, seguita da un processo di ricostruzione incrementale che stima progressivamente le pose delle camere e la struttura della scena. Il risultato di questo processo è tipicamente una *sparse point cloud* che rappresenta la struttura tridimensionale della scena.

Sebbene la SfM sia estremamente efficace nel ricostruire la struttura geometrica di una scena e nel stimare le pose delle camere, essa non fornisce una rappresentazione continua dell'aspetto visivo della scena né consente di generare direttamente viste fotorealistiche da punti di vista arbitrari. Questo limite ha portato allo sviluppo del *neural rendering*, un campo di ricerca che combina tecniche di computer grafica e machine learning per apprendere rappresentazioni continue delle scene tridimensionali direttamente dai dati visivi [4].

1.3 Neural Radiance Fields (NeRF)

L'innovazione più significativa nel campo del neural rendering è stata l'introduzione dei *Neural Radiance Fields* (NeRF) [5]. I NeRF appartengono alla famiglia delle **rappresentazioni implicite**, nelle quali la scena non è descritta tramite primitive geometriche esplicite ma è codificata all'interno di una funzione continua appresa da una rete neurale.

In questo approccio una scena tridimensionale viene modellata tramite una funzione volumetrica che associa a ogni coordinata spaziale tridimensionale e a ogni direzione di osservazione una densità volumetrica e un valore di *radiance*. Nel contesto della computer graphics, la radiance rappresenta la quantità di luce che viene emessa o riflessa da un punto della scena in una determinata direzione.

I NeRF utilizzano tecniche di **volume rendering**. Per ogni pixel dell'immagine viene tracciato un raggio nello spazio e vengono campionati numerosi punti lungo di esso; per ciascun punto la rete neurale restituisce la densità e il colore corrispondenti, che vengono successivamente integrati lungo il raggio per ottenere il colore finale del pixel.

Questo approccio consente di sintetizzare nuove viste della scena con un elevato livello di realismo e di modellare effetti di apparenza dipendenti dalla direzione di osservazione. Tuttavia, il grande numero di campionamenti necessari lungo ciascun raggio rende il processo di rendering **computazionalmente costoso**, limitando l'utilizzo diretto dei NeRF in applicazioni interattive o in tempo reale.

1.4 3D Gaussian Splatting

Per rispondere alla necessità di prestazioni compatibili con applicazioni interattive, Kerbl et al. hanno introdotto la tecnica del **3D Gaussian Splatting** [6]. Questo approccio rappresenta un ritorno a una **rappresentazione esplicita della scena**, basata su un insieme di **gaussiane tridimensionali anisotrope** che approssimano la distribuzione della radianza osservata nelle immagini di input [6].

Le gaussiane vengono inizializzate a partire dalla nuvola di punti sparsa generata tramite pipeline di **Structure-from-Motion** (SfM), dalla quale ereditano la struttura geometrica della scena [6]. Ogni gaussiana rappresenta una primitiva volumetrica dotata di parametri geometrici e radiometrici che possono essere ottimizzati durante il processo di *training*.

A differenza dei metodi basati su **Neural Radiance Fields**, che richiedono il campionamento di numerosi punti lungo ciascun raggio della camera tramite operazioni di *ray marching*, il **3D Gaussian Splatting** utilizza un algoritmo di rendering basato su **splatting**. In questo processo le gaussiane vengono proiettate sul piano immagine e composte tramite operazioni di rasterizzazione e *alpha blending*, consentendo una sintesi di nuove viste con elevata qualità visiva e prestazioni compatibili con il rendering in tempo reale [6].

Grazie a questa combinazione di rappresentazione esplicita e rendering efficiente su GPU, il **3D Gaussian Splatting** rappresenta oggi una delle tecniche più promettenti per la ricostruzione e la visualizzazione interattiva di ambienti tridimensionali complessi [6].

1.5 Obiettivi della tesi e panoramica dei risultati

Il presente lavoro di tesi analizza le potenzialità del **3D Gaussian Splatting** per la ricostruzione e la fruizione di ambienti tridimensionali in **contesti immersivi di realtà virtuale**. Nonostante la crescente diffusione di questa tecnica e la disponibilità di numerosi strumenti software dedicati, risultano ancora limitate analisi comparative sistematiche che considerino l'intero ecosistema degli strumenti disponibili e la loro integrazione in **pipeline operative per applicazioni immersive**.

L'obiettivo principale della tesi è quindi analizzare in modo sistematico gli strumenti disponibili per il Gaussian Splatting e valutare come le ricostruzioni generate con questa tecnica possano essere integrate in pipeline complete destinate alla visualizzazione in ambienti immersivi.

Per raggiungere questo obiettivo, il lavoro è stato articolato in tre fasi principali.

In una prima fase è stata realizzata una **overview comparativa di un insieme selezionato di strumenti rappresentativi** dell'ecosistema del Gaussian

Splatting, prendendo in considerazione sia i **trainer** utilizzati per la generazione delle ricostruzioni, sia i **visualizzatori** per la fruizione delle scene. Questa analisi ha permesso di confrontare le caratteristiche principali delle diverse soluzioni disponibili, evidenziandone le differenze.

Nella fase successiva è stata condotta una **analisi tecnica più approfondita** focalizzata sui **trainer open source** per la generazione delle ricostruzioni e sui **visualizzatori immersivi** per la fruizione delle scene in realtà virtuale. In questa fase sono state generate diverse ricostruzioni sperimentali tramite i trainer selezionati, successivamente analizzate attraverso strumenti di visualizzazione ed editing per valutare la qualità delle ricostruzioni, l'impatto delle operazioni di post-processing e le prestazioni di rendering nei visualizzatori XR.

Infine, sulla base dei risultati ottenuti è stata quindi selezionata una **pipeline operativa** composta da un **trainer open source** per la generazione delle ricostruzioni e da un **visualizzatore XR** per la loro fruizione in ambiente immersivo. La pipeline individuata è stata utilizzata per generare ricostruzioni tridimensionali e integrarle in un ambiente di realtà virtuale esplorabile tramite visore. L'ambiente immersivo ottenuto è stato utilizzato in uno **studio sperimentale con utenti**, finalizzato a valutare la *Quality of Experience* (QoE) delle ricostruzioni generate tramite 3D Gaussian Splatting quando visualizzate in un ambiente immersivo, con una particolare enfasi su tre dimensioni principali dell'esperienza utente: la qualità visiva percepita della ricostruzione, il senso di presenza nell'ambiente virtuale e il comfort dell'esperienza immersiva.

Le principali fasi del lavoro di tesi e la loro organizzazione all'interno della pipeline di analisi e sperimentazione sono riassunte nella Figura 1.1.

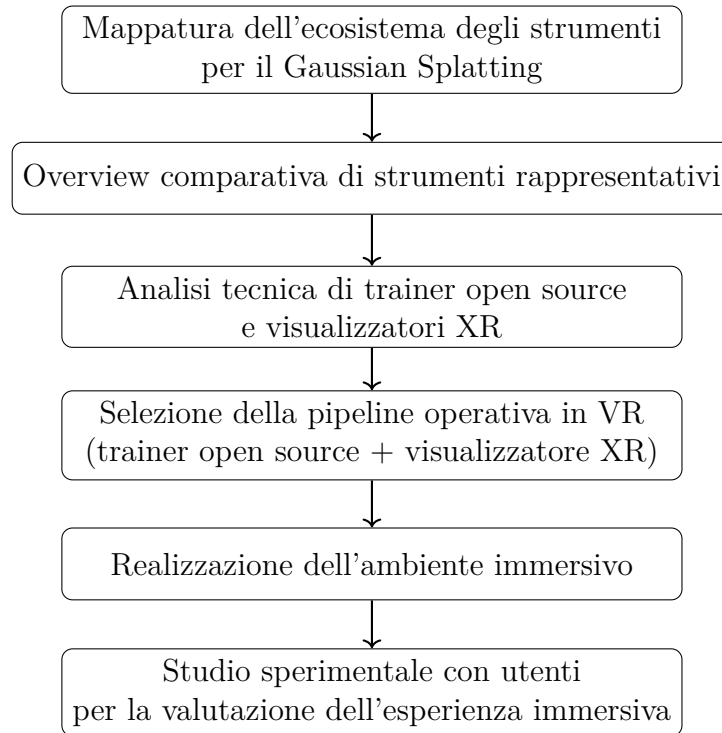


Figura 1.1: Workflow generale del lavoro di tesi: dall'analisi degli strumenti Gaussian Splatting alla selezione della pipeline operativa e alla valutazione sperimentale in ambiente immersivo.

1.6 Struttura della tesi

Il resto della tesi è organizzato come segue.

Il **Capitolo 2** presenta lo stato dell'arte relativo al **Gaussian Splatting** e agli strumenti sviluppati per la generazione e la visualizzazione di ricostruzioni tridimensionali basate su questa tecnica.

Il **Capitolo 3** descrive l'analisi comparativa condotta sugli strumenti di training e di visualizzazione.

Il **Capitolo 4** descrive l'integrazione della **pipeline** selezionata in un ambiente immersivo e presenta lo **studio sperimentale** condotto con utenti per valutare la qualità dell'esperienza.

Infine, il **Capitolo 5** riassume i risultati ottenuti e discute possibili sviluppi futuri.

Capitolo 2

Stato dell'arte

2.1 Structure-from-Motion e calibrazione delle camere

Molti metodi moderni di ricostruzione tridimensionale basati su immagini, tra cui Neural Radiance Fields e **3D Gaussian Splatting**, richiedono come informazioni di input le pose delle camere e una stima iniziale della struttura tridimensionale della scena. Queste informazioni vengono generalmente ottenute tramite tecniche di **Structure-from-Motion** (SfM), che consentono di ricostruire simultaneamente la geometria della scena e la configurazione delle camere a partire da un insieme di immagini acquisite da punti di vista differenti.

2.1.1 Structure-from-Motion

La **Structure-from-Motion** è una tecnica della computer vision che permette di stimare la struttura tridimensionale di una scena e le pose delle camere utilizzando un insieme di immagini bidimensionali. Il principio alla base della SfM consiste nello sfruttare le corrispondenze tra punti della scena osservati in immagini diverse per ricostruire la loro posizione nello spazio tridimensionale.

Le pipeline moderne di SfM seguono generalmente una sequenza di fasi ben definita. In primo luogo vengono estratte caratteristiche locali dalle immagini (*feature extraction*). Successivamente viene effettuata la ricerca delle corrispondenze tra immagini (*feature matching*), che consente di identificare punti della scena osservati da differenti punti di vista.

Una rappresentazione schematica delle principali fasi di una pipeline incrementale di Structure-from-Motion è riportata nella Figura 2.1.

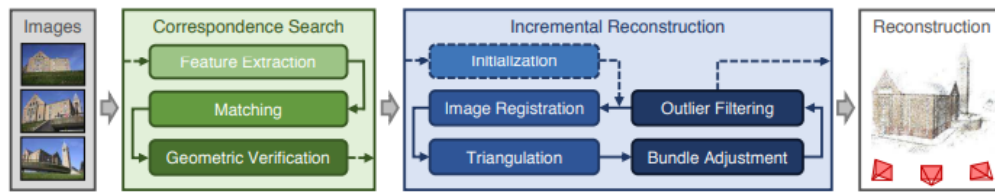


Figura 2.1: Pipeline incrementale di Structure-from-Motion. Il processo include l'estrazione delle feature, la ricerca delle corrispondenze tra immagini, la verifica geometrica e la ricostruzione incrementale della scena tramite registrazione delle immagini, triangolazione dei punti e *bundle adjustment*. Figura tratta da [3].

A partire da queste corrispondenze viene avviato un processo di ricostruzione incrementale che stima progressivamente le pose delle camere e la posizione dei punti nello spazio tridimensionale. Durante questo processo viene utilizzata una procedura di **bundle adjustment**, che ottimizza simultaneamente i parametri delle camere e le coordinate tridimensionali dei punti minimizzando l'errore di riproiezione tra i punti osservati nelle immagini e quelli ricostruiti [3].

Il risultato di questa procedura è tipicamente una **sparse point cloud**, che rappresenta una prima approssimazione della struttura tridimensionale della scena insieme alla stima delle pose delle camere.

2.1.2 COLMAP

Una delle implementazioni open source più diffuse delle pipeline **Structure-from-Motion** è il software **COLMAP** [3]. COLMAP implementa una **pipeline SfM incrementale** completa che include l'estrazione delle feature, il matching tra immagini, la stima delle pose delle camere e l'ottimizzazione tramite bundle adjustment.

Il sistema è progettato per gestire dataset di immagini non strutturati e fornisce strumenti per la ricostruzione automatica della scena a partire da fotografie acquisite con camere non calibrate. Tra le caratteristiche principali di COLMAP vi sono un sistema efficiente per la ricerca delle corrispondenze tra immagini e un algoritmo di ricostruzione incrementale robusto che consente di stimare progressivamente la struttura della scena.

Nel presente lavoro COLMAP è stato utilizzato per stimare le **pose delle camere** e generare una **nuvola di punti sparsa** della scena. Queste informazioni costituiscono l'inizializzazione necessaria per i metodi di ricostruzione basati su radiance fields, tra cui il **3D Gaussian Splatting**, che utilizzano le pose di camera e la struttura iniziale della scena come punto di partenza per l'ottimizzazione della rappresentazione tridimensionale.

2.2 3D Gaussian Splatting

Il **3D Gaussian Splatting** (3DGS) è una tecnica per la sintesi di nuove viste (*novel-view synthesis*) introdotta da Kerbl et al. [6] con l'obiettivo di combinare elevata qualità visiva e prestazioni compatibili con il rendering in tempo reale.

Il metodo introduce una rappresentazione esplicita dei campi di radianza basata su un insieme non strutturato di **gaussiane tridimensionali anisotrope**. Questa rappresentazione consente di modellare scene complesse mantenendo una struttura dei dati facilmente proiettabile e rasterizzabile sulla GPU.

Il framework proposto nel lavoro si basa su tre componenti principali: una rappresentazione esplicita della scena tramite **gaussiane 3D**, un processo di ottimizzazione che integra un meccanismo di **adaptive density control** e un **renderer differenziabile** progettato per supportare sia il training sia il rendering interattivo.

Grazie alla combinazione di questi elementi, il **3D Gaussian Splatting** consente di ottenere ricostruzioni di scene reali a partire da immagini multi-vista con qualità comparabile o superiore a molti metodi basati su radianza fields, come i Neural Radiance Fields e le loro varianti accelerate, mantenendo al contempo prestazioni sufficienti per il rendering in tempo reale [6].

2.2.1 Panoramica del metodo

La pipeline del **3D Gaussian Splatting** parte da un insieme di immagini multi-vista della scena e dalle relative pose di camera, tipicamente stimate tramite tecniche di **Structure-from-Motion** (SfM). Questo processo produce una **sparse point cloud** che fornisce una prima approssimazione della geometria della scena.

A partire da questa inizializzazione viene generato un insieme di **gaussiane tridimensionali** che costituiscono la rappresentazione esplicita della scena. Durante il training i parametri delle gaussiane vengono ottimizzati confrontando le immagini renderizzate con le immagini di riferimento.

Parallelamente all'ottimizzazione dei parametri, il metodo introduce un meccanismo di **adaptive density control** che modifica dinamicamente il numero e la distribuzione delle gaussiane nello spazio. Questo consente di aumentare la densità delle primitive nelle regioni che richiedono maggiore dettaglio e di eliminare gaussiane con contributo trascurabile.

Il rendering delle viste viene eseguito tramite un **rasterizzatore differenziabile** progettato per gestire lo *splatting* delle gaussiane anisotrope. Questo renderer è utilizzato durante il processo di ottimizzazione per generare le immagini sintetizzate confrontate con le immagini di training e, una volta completato il training, consente la sintesi di nuove viste con prestazioni compatibili con il rendering in tempo reale [6].

Il flusso complessivo del metodo e le principali fasi della pipeline sono illustrati nella Figura 2.2.

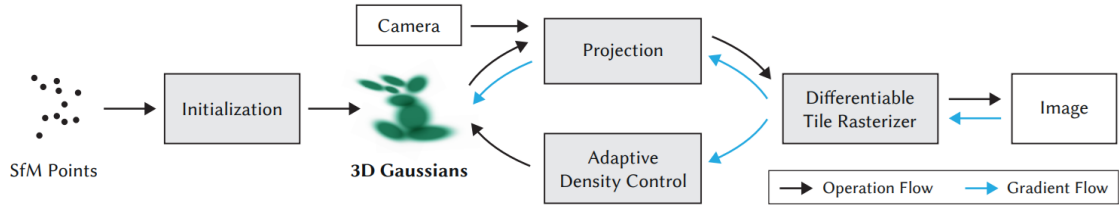


Figura 2.2: Pipeline del metodo 3D Gaussian Splatting. A partire da una nuvola di punti ottenuta tramite **Structure-from-Motion** (SfM) viene inizializzato un insieme di gaussiane tridimensionali. Durante il *training* i parametri delle gaussiane vengono ottimizzati e la loro densità adattata dinamicamente tramite **adaptive density control**. Il rendering delle viste avviene tramite un **rasterizzatore differenziabile** progettato per supportare il training e la visualizzazione in tempo reale. Figura tratta da [6].

2.2.2 Rappresentazione della scena tramite gaussiane anisotrope

Nel **3D Gaussian Splatting** la scena è rappresentata tramite un insieme di primitive gaussiane tridimensionali. Ogni gaussiana è caratterizzata da parametri geometrici e radiometrici che permettono di approssimare la distribuzione di radianza osservata nelle immagini di training.

A differenza di una gaussiana isotropa, una **gaussiana anisotropa** può assumere una forma ellissoidale orientata nello spazio tridimensionale. Questa proprietà consente alle primitive di adattarsi alla geometria della scena e di rappresentare efficacemente superfici sottili o strutture allungate.

Nel metodo proposto da Kerbl et al., ogni gaussiana è definita da una matrice di covarianza tridimensionale completa Σ , definita nello spazio della scena, e centrata in un punto (media) μ . La distribuzione gaussiana è espressa come:

$$G(x) = e^{-\frac{1}{2}x^T \Sigma^{-1} x} \quad (2.1)$$

dove x rappresenta la posizione nello spazio rispetto al centro della gaussiana.

Parametri geometrici

La matrice di covarianza Σ determina la forma e l'orientamento dell'ellissoide associato alla gaussiana. Durante il rendering, la covarianza viene trasformata nello spazio della camera secondo la relazione:

$$\Sigma' = JW\Sigma W^T J^T \quad (2.2)$$

dove W rappresenta la trasformazione di vista e J è il Jacobiano dell'approssimazione affine della trasformazione proiettiva [6].

Attributi radiometrici

L'aspetto direzionale della scena viene modellato tramite coefficienti di armoniche sferiche (**Spherical Harmonics**, SH), che consentono di descrivere variazioni di colore **dipendenti dalla direzione di osservazione**.

2.2.3 Ottimizzazione e controllo adattivo della densità

Il training del modello consiste in un processo iterativo in cui i parametri delle gaussiane vengono aggiornati per ridurre la differenza tra le immagini renderizzate e le immagini reali osservate.

Durante l'ottimizzazione viene applicato un meccanismo di **adaptive density control** che consente di modificare dinamicamente il numero di gaussiane nella scena.

In particolare, il metodo introduce due operazioni principali:

- **Cloning**: quando una regione della scena risulta sotto-rappresentata (*under-reconstruction*), una gaussiana viene duplicata per aumentare la densità delle primitive.
- **Splitting**: quando una gaussiana copre una regione troppo ampia (*over-reconstruction*), essa viene suddivisa in più gaussiane più piccole.

Parallelamente, gaussiane con contributo trascurabile possono essere rimosse tramite operazioni di **pruning**, mantenendo la rappresentazione compatta e concentrata nelle regioni più rilevanti della scena.

Una rappresentazione schematica delle operazioni di adaptive density control utilizzate durante il training del modello è riportata nella Figura 2.3.

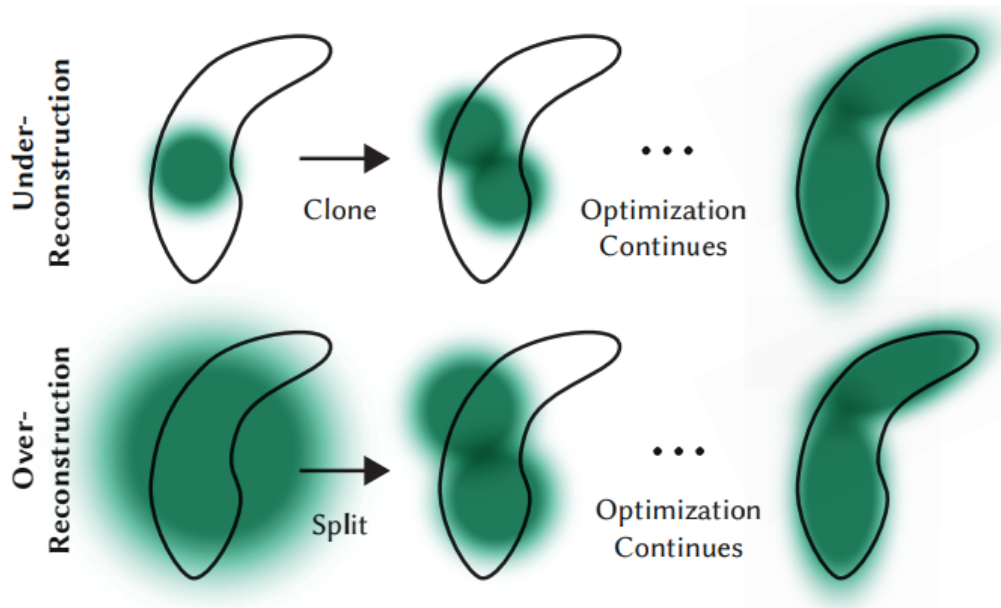


Figura 2.3: Schema del processo di **adaptive density control** nel 3D Gaussian Splatting. In presenza di sotto-ricostruzione della geometria una gaussiana può essere duplicata (clone), mentre in caso di over-reconstruction una gaussiana può essere suddivisa in più primitive (split). Figura tratta da [6].

2.2.4 Rendering differenziabile

Il rendering delle gaussiane avviene tramite un processo di **rasterizzazione differenziabile** progettato per essere efficiente su GPU. Le gaussiane vengono proiettate sul piano immagine e i loro contributi vengono combinati tramite operazioni di composizione.

Composizione tramite alpha blending

Ciascuna gaussiana include un parametro di opacità α che contribuisce alla formazione del colore finale dei pixel. Il colore risultante può essere espresso come:

$$C = \sum_i c_i \alpha_i \prod_{j < i} (1 - \alpha_j) \quad (2.3)$$

dove c_i rappresenta il colore della i -esima gaussiana.

Il renderer utilizzato nel **3D Gaussian Splatting** è inoltre **visibility-aware** e progettato per supportare un *backward pass* efficiente durante il training. Questo consente di propagare i gradienti attraverso il processo di rasterizzazione e di aggiornare direttamente i parametri delle gaussiane durante l'ottimizzazione [6].

2.2.5 Implementazione di riferimento

Insieme alla pubblicazione del metodo, gli autori hanno reso disponibile anche un'implementazione di riferimento del **3D Gaussian Splatting** [7]. Questa implementazione open source fornisce una pipeline completa per l'ottimizzazione e il rendering delle gaussiane tridimensionali a partire da immagini multi-vista e dalle pose delle camere ottenute tramite **Structure-from-Motion**.

Il codice rilasciato dagli autori costituisce oggi la base di riferimento per numerosi progetti successivi e ha contribuito alla rapida diffusione del metodo all'interno della comunità di ricerca e sviluppo. Diverse implementazioni e strumenti software sviluppati successivamente si basano infatti direttamente su questa pipeline o ne estendono alcune componenti, ad esempio introducendo miglioramenti nelle prestazioni, nella modularità del codice o nell'integrazione con framework di neural rendering esistenti.

Nel presente lavoro l'implementazione originale del **3D Gaussian Splatting** è stata utilizzata come punto di riferimento per comprendere la pipeline del metodo.

2.3 Ecosistema degli strumenti per 3D Gaussian Splatting

Dopo l'introduzione del metodo **3D Gaussian Splatting** e della relativa implementazione di riferimento, si è sviluppato rapidamente un ecosistema eterogeneo di strumenti software dedicati a diverse fasi del workflow. Tra questi rientrano strumenti per il training, come la pipeline **Splatfacto** integrata nel framework **Nerfstudio** o soluzioni commerciali come **Postshot**, sistemi di **visualizzazione**, tra cui il viewer **SIBR** sviluppato dal gruppo INRIA, strumenti di **editing** come **SuperSplat** e diverse **integrazioni per game engine** che consentono di utilizzare scene rappresentate tramite gaussiane all'interno di motori grafici come **Unity**, **Unreal Engine** e **PlayCanvas**. Una panoramica aggiornata di questo ecosistema è disponibile in raccolte curate dalla comunità, come il repository *awesome-3d-gaussian-splatting* [8].

2.4 Trainer per 3D Gaussian Splatting

All'interno dell'ecosistema **Gaussian Splatting** sono stati sviluppati numerosi strumenti software dedicati al training di modelli basati su gaussiane tridimensionali. Questi sistemi implementano procedure di ottimizzazione che permettono di ricostruire una scena tridimensionale a partire da dati acquisiti nel mondo reale.

Le pipeline di training possono includere diverse fasi di preparazione dei dati, come l'estrazione di frame da sequenze video, la stima delle pose delle camere tramite

tecniche di Structure-from-Motion e la costruzione di una prima approssimazione della geometria della scena. Il grado di integrazione di queste fasi varia tuttavia tra i diversi strumenti: alcune soluzioni includono l'intero processo di ricostruzione, mentre altre richiedono dataset già preprocessati.

I trainer disponibili possono essere suddivisi in due categorie principali: **strumenti commerciali**, progettati per workflow automatizzati e orientati alla produzione di contenuti, e **implementazioni open source**, utilizzate principalmente in ambito di ricerca e sperimentazione.

2.4.1 Trainer commerciali

I **trainer commerciali** sono generalmente progettati per offrire workflow integrati che consentono di generare modelli basati su **Gaussian Splatting** a partire da acquisizioni della scena reale o da dataset di immagini già disponibili. Tali strumenti integrano diverse fasi della ricostruzione tridimensionale, tra cui la preparazione dei dati, la stima delle pose delle camere e l'ottimizzazione della rappresentazione della scena basata su gaussiane.

Uno degli strumenti più completi in questo ambito è **Postshot**, sviluppato da Jawset. Postshot implementa una pipeline *end-to-end* per la ricostruzione di scene tridimensionali tramite **Gaussian Splatting**. Il software consente di generare modelli direttamente a partire da sequenze video o da insiemi di immagini, gestendo internamente diverse fasi della ricostruzione, tra cui la stima delle pose delle camere e la generazione di una prima struttura geometrica della scena. Il processo di *training* può essere avviato importando un video o una cartella di immagini, dai quali il sistema ricostruisce progressivamente la scena tridimensionale e ottimizza la rappresentazione basata su gaussiane [9]. Una volta completata la procedura, il modello risultante può essere utilizzato per il rendering di nuove viste o esportato in formati compatibili con altri strumenti di visualizzazione.

Un approccio differente è rappresentato da **Niantic Capture**, evoluzione dell'applicazione mobile **Scaniverse**. Questa piattaforma consente di acquisire dati tridimensionali direttamente da dispositivi mobili e di generare rappresentazioni spaziali con supporto ai formati basati su **Gaussian Splatting**. Il sistema è progettato principalmente per la raccolta e la condivisione di dati spaziali destinati ad applicazioni di realtà aumentata e sistemi di comprensione dell'ambiente.

Un'altra soluzione è offerta da **Polycam**, che mette a disposizione strumenti web e mobile per la generazione di modelli **Gaussian Splatting**. La piattaforma permette sia di acquisire nuovi dati tramite dispositivi mobili, ad esempio registrando video con uno smartphone, sia di caricare dataset esistenti per ottenere una ricostruzione tridimensionale della scena [10].

Luma AI rappresenta una piattaforma cloud per la generazione di scene tridimensionali navigabili a partire da video o sequenze di immagini. Il sistema esegue

automaticamente le diverse fasi della ricostruzione e produce una rappresentazione interattiva della scena che può essere esplorata da nuovi punti di vista tramite il visualizzatore web della piattaforma [11].

KIRI Engine è un'altra soluzione commerciale che integra funzionalità di acquisizione e ricostruzione tridimensionale. Oltre alle tradizionali pipeline di scansione fotogrammetrica, il sistema supporta la generazione di modelli basati su 3D Gaussian Splatting. L'applicazione consente di creare modelli 3DGS direttamente dall'ambiente di acquisizione e di applicare operazioni di post-elaborazione, come il ritaglio della scena o la rimozione di elementi indesiderati, prima dell'esportazione del modello [12].

Infine, la piattaforma **Varjo Teleport** è progettata per la cattura e la ricostruzione di ambienti reali destinati ad applicazioni professionali di *Extended Reality* (XR). Il sistema utilizza una pipeline cloud per trasformare dataset fotografici o video in modelli tridimensionali fotorealistici [13]. I modelli generati possono essere recuperati tramite API e integrati in flussi di lavoro più complessi, come simulazioni industriali o ambienti di addestramento in realtà virtuale e mista.

I diversi strumenti commerciali si distinguono principalmente per il tipo di workflow supportato. **Postshot** rappresenta una soluzione orientata al *training* desktop di modelli **Gaussian Splatting**, mentre strumenti come **Polycam**, **Niantic Capture** e **KIRI Engine** privilegiano pipeline di acquisizione tramite dispositivi mobili. Piattaforme come **Luma AI** e **Varjo Teleport** sono invece concepite come sistemi cloud per la generazione e la distribuzione di scene tridimensionali interattive.

2.4.2 Trainer open source

Accanto alle soluzioni commerciali, l'ecosistema del **3D Gaussian Splatting** comprende numerose implementazioni **open source** sviluppate principalmente in ambito accademico e di ricerca. Questi strumenti sono generalmente progettati per offrire maggiore flessibilità nella sperimentazione di nuove tecniche di training, nell'integrazione con pipeline personalizzate e nello sviluppo di varianti del metodo originale.

A differenza di molte piattaforme commerciali, le implementazioni open source possono richiedere dataset già preprocessati o una configurazione più esplicita delle diverse fasi della pipeline di ricostruzione. Ciò consente tuttavia un controllo più diretto sui parametri di training e sulla struttura della rappresentazione della scena.

L'implementazione di riferimento è quella pubblicata dal gruppo **GraphDeco dell'INRIA**, sviluppata in concomitanza con il lavoro originale di Kerbl et al. [14]. Questa implementazione riproduce la pipeline descritta nel paper e include la rappresentazione della scena tramite gaussiane tridimensionali anisotrope, il meccanismo di adaptive density control e il rasterizzatore differenziabile progettato

per il rendering efficiente su GPU. Il repository fornisce inoltre strumenti per la preparazione dei dataset, che permettono di eseguire automaticamente le fasi di preprocessing basate su **Structure-from-Motion** tramite **COLMAP** e di generare i dati necessari per l'avvio del training.

Nel tempo sono stati sviluppati framework più modulari progettati per facilitare la ricerca e l'integrazione con altre pipeline di neural rendering. In questo contesto si colloca **gsplat**, una libreria sviluppata nell'ambito del progetto **Nerfstudio** che implementa un rasterizzatore **Gaussian Splatting** altamente ottimizzato basato su CUDA [15]. Questa libreria fornisce un backend efficiente per il rendering e il training di scene rappresentate tramite gaussiane tridimensionali ed è progettata per essere integrata in pipeline di training basate su framework come **PyTorch**.

Un esempio rilevante di framework che utilizza questa libreria è **Nerfstudio**, un ambiente **open source** per la ricerca nel campo del neural rendering che supporta diversi metodi di rappresentazione della scena [16]. All'interno di questo framework è stata introdotta la pipeline **Splatfacto**, dedicata al training di modelli basati su **Gaussian Splatting** e costruita proprio sul backend fornito da **gsplat**. Nerfstudio mette inoltre a disposizione strumenti per la gestione dei dataset, la configurazione delle pipeline di training e la visualizzazione dei risultati.

Accanto a questi strumenti più diffusi sono state sviluppate diverse implementazioni alternative con obiettivi differenti. Il progetto **OpenSplat** propone un'implementazione **open source** della pipeline di **Gaussian Splatting** progettata con un'architettura modulare, pensata per facilitare la sperimentazione e lo sviluppo di varianti del metodo originale [17].

Analogamente, **LichtFeld Studio** rappresenta un ambiente **open source** progettato per il training e la gestione di scene basate su **Gaussian Splatting** [18]. Il sistema è sviluppato con particolare attenzione alle prestazioni e all'estensibilità e fornisce una pipeline completa per la generazione e l'elaborazione di modelli basati su gaussiane tridimensionali. A differenza di molte altre implementazioni orientate esclusivamente alla ricerca, LichtFeld Studio include anche un'interfaccia utente grafica che consente di gestire dataset, configurare il training e visualizzare le scene ricostruite in modo interattivo.

Altri progetti esplorano invece l'utilizzo di framework di calcolo alternativi. L'implementazione **Taichi 3D Gaussian Splatting**, ad esempio, utilizza il linguaggio **Taichi** per realizzare una versione della pipeline GS con particolare attenzione alla portabilità e alla gestione efficiente delle operazioni su GPU [19]. Un approccio simile è adottato dal progetto **Warp 3DGS**, che propone un'implementazione sperimentale basata su **Warp**, un framework per la scrittura di kernel paralleli accelerati su GPU.

Infine, alcune iniziative si concentrano sull'integrazione del **Gaussian Splatting** con architetture di sistema più ampie. Il progetto **Grendel-GS**, sviluppato in ambito accademico, propone un sistema progettato per scalare il training di modelli

Gaussian Splatting su infrastrutture di calcolo più complesse [20].

Nel complesso, la presenza di numerose implementazioni **open source** testimonia il rapido sviluppo dell'ecosistema **Gaussian Splatting** e il forte interesse della comunità di ricerca verso questa tecnica. Tali strumenti costituiscono una base fondamentale per l'esplorazione di nuove varianti del metodo e per l'integrazione del **Gaussian Splatting** in pipeline di ricostruzione tridimensionale sempre più articolate.

2.5 Visualizzatori, editor e integrazioni per 3D Gaussian Splatting

Oltre agli strumenti dedicati al training dei modelli, l'ecosistema del **3D Gaussian Splatting** comprende numerosi strumenti progettati per la **visualizzazione**, l'**ispezione**, l'**editing** e l'**integrazione** delle scene ricostruite. Questi strumenti consentono di esplorare i modelli gaussiani generati dalle pipeline di training e di utilizzarli in contesti applicativi differenti, come visualizzazione interattiva o ambienti immersivi.

A differenza dei trainer, che implementano procedure di ottimizzazione della rappresentazione della scena, i visualizzatori operano generalmente **dopo la fase di training** e sono progettati per il **rendering** delle gaussiane tridimensionali. Nel tempo si è sviluppato un ecosistema eterogeneo di strumenti che include **visualizzatori standalone**, **editor** per scene composte da gaussiane, **integrazioni con motori grafici** e sistemi dedicati alla **visualizzazione immersiva in realtà virtuale**.

2.5.1 Visualizzatori

Tra gli strumenti più diffusi per la visualizzazione di scene **Gaussian Splatting** vi sono diversi visualizzatori progettati per l'esplorazione interattiva delle ricostruzioni tridimensionali.

Il **SIBR Viewer**, sviluppato dal gruppo **GraphDeco** dell'INRIA, è il visualizzatore distribuito insieme all'implementazione originale del metodo **3D Gaussian Splatting** [14]. Il viewer è basato sul framework **SIBR** (Simple Interactive Bundle-based Rendering) e consente sia la visualizzazione di modelli già addestrati sia il monitoraggio del processo di training tramite una modalità di visualizzazione remota [21].

Un altro strumento ampiamente utilizzato è il **Nerfstudio Viewer**, integrato nel framework **Nerfstudio** [16]. Questo visualizzatore permette la visualizzazione

interattiva della scena durante il processo di training tramite un'interfaccia accessibile da browser, consentendo di esplorare in tempo reale l'evoluzione della ricostruzione [22].

Nel contesto dei visualizzatori web è disponibile anche **SuperSplat Viewer**, sviluppato nell'ecosistema **PlayCanvas**. Questo visualizzatore consente di caricare e visualizzare modelli generati tramite **Gaussian Splatting** direttamente all'interno di un browser tramite tecnologie *WebGL* [23].

Un ulteriore esempio è rappresentato dal progetto **splat**, un visualizzatore *WebGL* open source che permette la navigazione interattiva di scene **Gaussian Splatting** direttamente nel browser senza richiedere software dedicato [24].

2.5.2 Strumenti di editing

Oltre alla visualizzazione delle scene, alcuni strumenti forniscono funzionalità di **editing** che consentono di modificare direttamente le rappresentazioni gaussiane.

Tra questi strumenti rientra **SuperSplat**, un editor web open source che permette di caricare modelli generati tramite **Gaussian Splatting**, ispezionarne la struttura e applicare operazioni di modifica tramite interfaccia grafica [25].

Un altro strumento dedicato all'editing è **GaussianEditor**, un sistema progettato per la manipolazione controllata di scene generate con il **Gaussian Splatting** che consente di modificare direttamente la struttura della rappresentazione tridimensionale [26].

2.5.3 Integrazioni con game engine

Un'altra categoria di strumenti riguarda le **integrazioni con game engine** utilizzati nello sviluppo di applicazioni interattive. Questi strumenti consentono di utilizzare modelli generati tramite **Gaussian Splatting** all'interno di pipeline di rendering in tempo reale e ambienti di sviluppo per applicazioni grafiche.

Un esempio è rappresentato da **UnityGaussianSplatting**, sviluppato da Prancišius, che fornisce un renderer per la visualizzazione di modelli generati con il **Gaussian Splatting** all'interno del motore grafico **Unity**. Il sistema permette di importare modelli gaussiani come *asset* e di integrarli direttamente nelle scene **Unity** [27].

Nel contesto delle applicazioni web, il progetto **GaussianSplats3D** propone un renderer *WebGL* progettato per l'integrazione di modelli generati con il **Gaussian Splatting** in applicazioni basate sul motore grafico **PlayCanvas** e in altri ambienti di rendering *browser-based* [28].

Sono inoltre disponibili integrazioni sperimentali per **Unreal Engine**. Un esempio è rappresentato dal plugin **UEGaussianSplatting**, che consente di visualizzare

modelli generati tramite **Gaussian Splatting** all'interno della pipeline grafica del motore [29].

2.5.4 Visualizzatori XR

Infine, alcuni strumenti sono progettati specificamente per la visualizzazione immersiva delle scene **Gaussian Splatting** in ambienti di **realtà virtuale**.

Il progetto **Unity-VR-Gaussian-Splatting**, sviluppato da **ninjamode**, estende l'integrazione di modelli generati con il **Gaussian Splatting** nel *game engine* **Unity** introducendo il supporto alla visualizzazione in ambienti XR [30].

Il progetto **GaussianSplattingVRViewerUnity**, sviluppato dal gruppo **Clar-té**, propone invece un visualizzatore VR completo basato su **Unity** progettato per l'esplorazione immersiva di modelli generati tramite **Gaussian Splatting** [31].

Nel complesso, gli strumenti disponibili per la visualizzazione di scene generate col **Gaussian Splatting** coprono un ampio spettro di applicazioni, che spaziano dalla semplice esplorazione delle ricostruzioni tridimensionali fino all'integrazione in ambienti di rendering interattivo e sistemi di realtà virtuale.

2.6 Sintesi degli strumenti analizzati

La Tabella 2.1 riassume gli strumenti analizzati nello stato dell'arte e le loro principali caratteristiche. Gli strumenti sono classificati in base alla loro funzione principale all'interno dell'ecosistema del **3D Gaussian Splatting**, distinguendo tra **trainer**, **visualizzatori**, **editor** e strumenti di **integrazione con game engine**. Questa panoramica costituisce la base per l'analisi comparativa presentata nel Capitolo 3.

Tabella 2.1: Panoramica degli strumenti per **3D Gaussian Splatting** analizzati nello stato dell'arte

Strumento	Categoria	Tipo	Piattaforma
Postshot	Trainer	Commerciale	Desktop
Niantic Capture	Trainer	Commerciale	Mobile / Cloud
Polycam	Trainer	Commerciale	Mobile / Web
Luma AI	Trainer	Commerciale	Cloud
KIRI Engine	Trainer	Commerciale	Mobile / Desktop
Varjo Teleport	Trainer	Commerciale	Cloud
GraphDeco Gaussian Splatting	Trainer	Open source	Desktop
Nerfstudio / Splatfacto	Trainer	Open source	Desktop
OpenSplat	Trainer	Open source	Desktop
LichtFeld Studio	Trainer	Open source	Desktop
Taichi 3D Gaussian Splatting	Trainer	Open source	Desktop
Grendel-GS	Trainer	Open source	Research prototype
SIBR Viewer	Visualizzatore	Open source	Desktop
Nerfstudio Viewer	Visualizzatore	Open source	Web
SuperSplat Viewer	Visualizzatore	Open source	Web
splat	Visualizzatore	Open source	Web
SuperSplat	Editor	Open source	Web
GaussianEditor	Editor	Research prototype	Desktop
Unity GaussianSplatting	Integrazione	Open source	Unity
GaussianSplats3D	Integrazione	Open source	Web / PlayCanvas
UEGaussianSplatting	Integrazione	Research prototype	Unreal Engine
Unity-VR-Gaussian-Splatting	Viewer XR	Open source	Unity VR
GaussianSplatting	Viewer XR	Open source	Unity VR
VRViewer Unity			

2.7 Selezione degli strumenti per l'analisi comparativa

L'analisi dello stato dell'arte evidenzia come l'ecosistema degli strumenti per il **3D Gaussian Splatting** sia caratterizzato da una notevole eterogeneità. Sono infatti disponibili numerose soluzioni software che coprono diverse fasi del workflow, tra cui il **training** delle ricostruzioni, la **visualizzazione e l'editing** delle scene generate e la loro **integrazione** all'interno di *game engine* o ambienti immersivi.

Considerato il numero crescente di strumenti disponibili e le differenze tra le diverse pipeline proposte, è stato necessario selezionare un insieme più ristretto di strumenti **rappresentativi** da sottoporre a un'analisi comparativa più dettagliata.

Sono stati quindi selezionati i migliori strumenti appartenenti alle principali categorie identificate nello stato dell'arte, includendo sia sistemi dedicati al **training** delle ricostruzioni sia strumenti per la **visualizzazione**, l'editing e l'integrazione delle scene basate su 3D Gaussian Splatting all'interno di *game engine*.

In particolare, per quanto riguarda i trainer sono stati considerati:

- **Luma AI**
- **Varjo Teleport**
- **Polycam**
- **Postshot**
- **Scaniverse**
- **Inria gaussian-splatting**
- **gsplat**
- **Nerfstudio**
- **OpenSplat**
- **LichtFeld Studio**

Per quanto riguarda invece gli strumenti di visualizzazione, editing e integrazione sono stati presi in considerazione:

- **SuperSplat**
- **Nerfstudio Viewer**
- **SIBR Viewer**
- **UnityGaussianSplatting**
- **Unity-VR-Gaussian-Splatting**
- **GaussianSplattingVRViewerUnity**

Questi strumenti costituiscono l'insieme di tecnologie analizzato nel capitolo successivo, nel quale viene presentata un'analisi comparativa delle loro caratteristiche.

L'analisi si concentra in modo più approfondito sui **trainer open source** per la generazione delle ricostruzioni e sui **visualizzatori XR** per la fruizione delle scene in ambienti immersivi.

Capitolo 3

Analisi comparativa

3.1 Metodologia

L'analisi comparativa presentata in questo capitolo segue una struttura metodologica organizzata su due livelli distinti ma complementari: una **comparazione di overview dei tool** e una **analisi tecnica controllata**.

La sezione di overview documenta e confronta qualitativamente una selezione di strumenti appartenenti all'ecosistema del 3D Gaussian Splatting, includendo trainer commerciali, trainer open source, visualizzatori desktop, visualizzatori web-based, strumenti di integrazione dei modelli generati tramite Gaussian Splatting all'interno di game engine e sistemi di visualizzazione immersiva. Questa parte fornisce una panoramica comparativa delle caratteristiche principali dei tool.

L'analisi tecnica è invece limitata a un sottoinsieme di strumenti selezionati ed è stata condotta in condizioni controllate e riproducibili. In particolare, l'analisi si concentra su:

- trainer open source per la generazione delle ricostruzioni
- visualizzatori e tool di integrazione nel *game engine* Unity XR per la visualizzazione immersiva

3.1.1 Pipeline di analisi tecnica

L'analisi tecnica segue una pipeline che riflette l'intero *workflow* di generazione, analisi e visualizzazione delle ricostruzioni basate su 3D Gaussian Splatting.

La pipeline adottata è composta dalle seguenti fasi:

1. training delle ricostruzioni tramite trainer open source
2. analisi quantitativa e analisi visiva delle ricostruzioni *raw*

3. operazioni di post-processing finalizzate alla pulizia delle gaussiane
4. analisi quantitativa dell'impatto delle operazioni di post-processing
5. analisi visiva delle ricostruzioni pulite
6. confronto qualitativo delle ricostruzioni negli strumenti per la visualizzazione XR
7. selezione del sistema di visualizzazione immersiva di riferimento
8. valutazione delle prestazioni di rendering nel sistema selezionato

Dopo la fase di training è stata eseguita una prima analisi quantitativa dei file di output prodotti dai diversi trainer, esportati in formato `.ply`. Questa analisi ha permesso di caratterizzare la struttura delle ricostruzioni generate misurando parametri come il **numero di gaussiane** presenti nella scena e la **dimensione del file** risultante.

Successivamente è stata effettuata un'analisi visiva delle ricostruzioni raw, con l'obiettivo di osservare la struttura delle scene generate prima di qualsiasi operazione di modifica.

Le ricostruzioni sono state quindi sottoposte a operazioni di pulizia tramite strumenti di visualizzazione ed editing desktop. Per valutare l'impatto di queste operazioni è stata eseguita una nuova analisi quantitativa dei file `.ply`, confrontando i valori ottenuti prima e dopo il post-processing in termini di **numero di gaussiane** e **dimensione del file**.

È stata inoltre condotta un'analisi visiva delle ricostruzioni dopo il post-processing al fine di osservare come le operazioni di pulizia influissero sulla qualità percepita delle scene e sulla presenza di eventuali artefatti.

Le ricostruzioni risultanti sono state successivamente utilizzate per una comparazione qualitativa negli strumenti di visualizzazione XR. In questa fase sono state considerate esclusivamente le ricostruzioni della scena **indoor**, al fine di mantenere costante il contenuto della scena durante l'esplorazione in realtà virtuale.

Sulla base di questa comparazione è stato quindi selezionato il sistema ritenuto più adatto alla visualizzazione immersiva delle ricostruzioni Gaussian Splatting. Le prestazioni di rendering sono state quindi analizzate utilizzando questo sistema di riferimento e considerando come input le ricostruzioni prodotte da tutti i trainer.

Questa organizzazione consente di analizzare separatamente:

- la struttura e la densità delle ricostruzioni generate
- la qualità delle ricostruzioni raw
- l'impatto delle operazioni di pulizia delle gaussiane

- la qualità delle ricostruzioni dopo il post-processing
- il comportamento delle scene nei sistemi di visualizzazione XR
- le prestazioni di rendering durante l'esplorazione immersiva

La pipeline utilizzata per l'analisi tecnica degli strumenti e per la selezione della pipeline operativa è illustrata nella Figura 3.1.

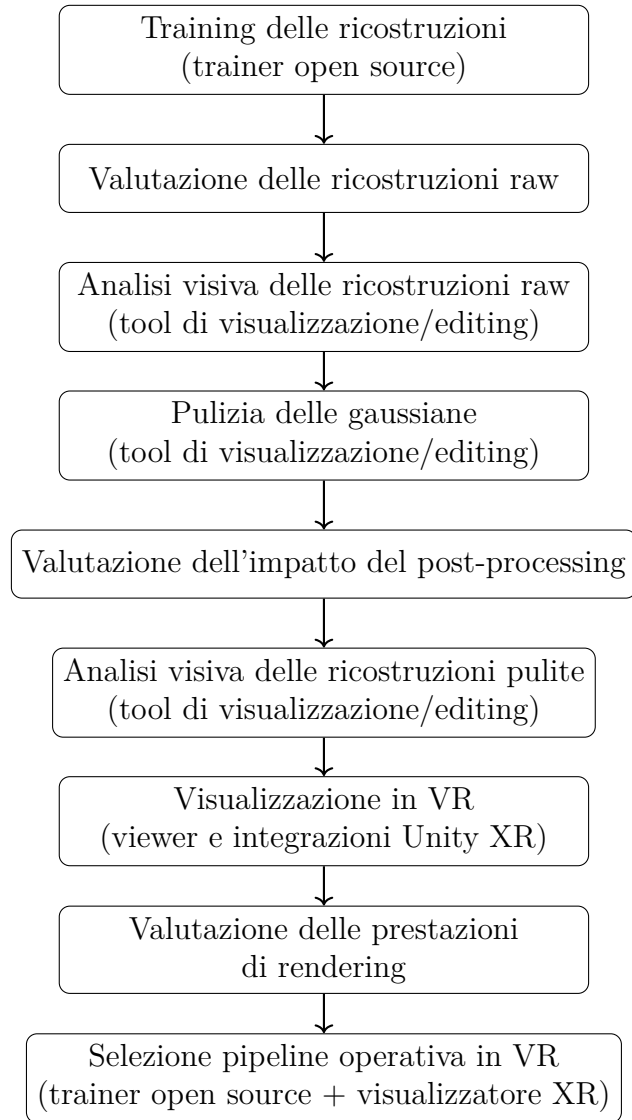


Figura 3.1: Pipeline sperimentale utilizzata per l'analisi comparativa degli strumenti Gaussian Splatting e la selezione della pipeline operativa da utilizzare in ambiente immersivo.

3.1.2 Condizioni sperimentali

Tutte le analisi sperimentali sono state condotte utilizzando condizioni di esecuzione controllate e configurazioni coerenti tra i diversi tool, al fine di garantire la comparabilità dei risultati.

Per il training delle ricostruzioni sono stati utilizzati due dataset video rappresentativi di scenari differenti: una scena **indoor** e una scena **outdoor**. I video sono stati selezionati dal dataset *DL3DV-10K-Sample* disponibile su HuggingFace [32].

Le ricostruzioni generate dai trainer sono state esportate in formato `.ply`, utilizzato come base per tutte le successive analisi quantitative e qualitative.

Le ricostruzioni sono state inizialmente analizzate nella loro forma originale (raw) e successivamente sottoposte a operazioni di pulizia tramite un visualizzatore desktop con funzionalità di editing. Le stesse operazioni di post-processing sono state applicate a tutte le ricostruzioni per mantenere condizioni di confronto uniformi.

Le ricostruzioni pulite sono state infine utilizzate come input per i sistemi di visualizzazione XR. In questa fase sono state considerate esclusivamente le ricostruzioni della scena **indoor**, al fine di mantenere condizioni di confronto coerenti tra i diversi sistemi di visualizzazione immersiva.

3.1.3 Hardware

Tutte le analisi sono state eseguite utilizzando la stessa configurazione hardware e software, al fine di garantire la comparabilità dei risultati ottenuti.

Il sistema utilizzato è costituito da un PC con sistema operativo **Windows 11**, equipaggiato con una GPU **NVIDIA RTX 4060**. Per la valutazione delle prestazioni in ambiente immersivo è stato utilizzato un visore **Meta Quest 3**.

3.1.4 Metriche di valutazione

Per il benchmark dei trainer open source sono state considerate metriche quantitative derivate direttamente dai file di output esportati in formato `.ply`.

Le seguenti metriche primarie sono state misurate per ciascun trainer:

- **Output size (MB)**, ovvero la dimensione del file del modello generato
- **Gaussian count**, cioè il numero di gaussiane presenti nella scena
- **Training time (min)**, ossia il tempo necessario per completare il training

A partire da queste grandezze sono state inoltre calcolate metriche normalizzate utilizzate per confrontare l'efficienza delle diverse pipeline:

- **Storage per 100k Gaussians (MB)**
- **Training time per 100k Gaussians (min)**

Per l'analisi dell'impatto del post-processing sono state invece considerate esclusivamente le metriche strutturali delle ricostruzioni, ovvero la dimensione del modello e il numero di gaussiane, confrontate prima e dopo le operazioni di pulizia.

Accanto alle metriche quantitative è stata inoltre condotta un'analisi visiva delle ricostruzioni, sia nella loro forma raw sia dopo il post-processing.

Per quanto riguarda i sistemi di visualizzazione XR, la valutazione si concentra sulle prestazioni di rendering durante l'esplorazione immersiva. In questo contesto vengono misurati:

- **Average FPS**
- **Average frame time (ms)**
- **Average maximum frame time (ms)**

Le misure sono state acquisite come media di due esecuzioni, utilizzando una finestra di acquisizione di 30 secondi successiva a un tempo di caricamento iniziale di 5 secondi.

3.2 Overview degli strumenti

Prima dell'analisi tecnica è stata condotta una **comparazione di overview** di alcuni strumenti per il 3D Gaussian Splatting. Questa fase ha lo scopo di analizzare qualitativamente le caratteristiche principali dei tool considerati, le modalità di utilizzo e il loro ruolo nelle diverse fasi del workflow di generazione, ispezione e visualizzazione delle ricostruzioni.

L'analisi è basata sulle informazioni riportate nella documentazione ufficiale dei progetti e su osservazioni derivate dall'utilizzo diretto degli strumenti.

Gli strumenti analizzati sono stati organizzati in due gruppi principali:

- **trainer**, utilizzati per la generazione delle ricostruzioni Gaussian Splatting a partire da immagini o video;
- **visualizzatori, editor e strumenti di integrazione**, utilizzati per l'ispezione, la modifica e la visualizzazione dei modelli generati, nonché per la loro integrazione in ambienti di rendering come Unity.

3.2.1 Trainer

Questa sezione presenta una comparazione qualitativa di diversi **trainer per Gaussian Splatting**. I tool considerati includono sia piattaforme commerciali sia implementazioni open source e sono stati analizzati sulla base della documentazione ufficiale e di osservazioni pratiche relative alle modalità di utilizzo.

L'analisi prende in considerazione diversi aspetti dei sistemi di training, tra cui la piattaforma di esecuzione, i formati di input e output supportati e la complessità di utilizzo.

I trainer considerati sono stati suddivisi in due gruppi principali: **soluzioni commerciali** e **implementazioni open source**.

Trainer commerciali

Le Tabelle 3.1 e 3.2 riassumono le principali caratteristiche dei trainer commerciali considerati nell'analisi.

Tabella 3.1: Informazioni generali sui trainer commerciali.

Tipo	Nome	Tag	Open Source	Note
Trainer + Visualizzatore	Luma AI	trainer, online, commerciale, web-view	No	Pipeline completamente automatizzata basata su cloud
Trainer + Visualizzatore	Teleport	trainer, online, commerciale, immersive-view	No	Pipeline cloud orientata alla visualizzazione desktop e VR
Trainer + Visualizzatore	Polycam	trainer, online, mobile, commerciale	No	Pipeline GS disponibile su dispositivi mobili e cloud
Trainer + Visualizzatore	Postshot	trainer, desktop, commerciale	No	Training GS eseguito localmente su desktop; esportazione .ply disponibile solo nei piani a pagamento
Trainer + Visualizzatore	Scaniverse	trainer, mobile, commerciale	No	Training GS eseguito direttamente sul dispositivo con esportazione .ply

Tabella 3.2: Caratteristiche delle pipeline di training dei trainer commerciali.

Nome	Input	Formato Output	Dim. Output	Tempo	Livello di accessibilità
Luma AI	Video / Immagini	Proprietario, .ply	Bassa	Medio	Facile
Teleport	Video / Immagini	Proprietario	Variabile	Medio	Facile
Polycam	Video / Immagini	Formati multipli (.ply a pagamento)	Media	Basso	Facile
Postshot	Immagini / Video	.ply, .spz, .psht	Alta	Medio	Medio
Scaniverse	Video	.ply, .spz	Media	Basso	Facile

Tra le soluzioni commerciali analizzate emergono differenze rilevanti principalmente nel tipo di workflow supportato e nel livello di automazione delle pipeline di training.

Luma AI e **Teleport** adottano un approccio basato su infrastruttura cloud, nel quale l'intero processo di ricostruzione viene eseguito automaticamente a partire da dataset di immagini o video caricati sulla piattaforma. In entrambi i casi il livello di difficoltà iniziale risulta relativamente basso, mentre i tempi di elaborazione dipendono dalle risorse di calcolo disponibili nel servizio cloud. Teleport prevede inoltre alcune limitazioni nel piano gratuito, tra cui un numero massimo di ricostruzioni che possono essere generate senza abbonamento.

Polycam e **Scaniverse** sono invece strumenti fortemente orientati all'acquisizione tramite dispositivi mobili. Il workflow è progettato per essere rapido e facilmente accessibile, con tempi di elaborazione generalmente contenuti. Nel caso di Polycam, tuttavia, l'utilizzo della pipeline Gaussian Splatting tramite interfaccia web richiede un piano a pagamento, sebbene sia disponibile un periodo di prova gratuito.

Postshot rappresenta una soluzione differente, in quanto implementa una pipeline di training locale eseguita direttamente su desktop. Questo approccio consente un controllo maggiore sui parametri di ricostruzione ma comporta generalmente tempi di elaborazione più elevati e modelli di dimensioni maggiori rispetto alle soluzioni cloud o mobile. Va inoltre sottolineato che l'esportazione dei modelli in formato .ply è disponibile soltanto nelle versioni a pagamento del software.

Trainer open source

Le Tabelle 3.3 e 3.4 riassumono le principali caratteristiche dei trainer open source considerati nell'analisi.

Tabella 3.3: Informazioni generali sui trainer open source.

Tipo	Nome	Tag	Open Source	Piattaforma	Note
Trainer	Inria gaussian-splatting	trainer, desktop, open-source	Sì	Windows / Linux	Implementazione open source di riferimento
Trainer	gsplat	trainer, backend, desktop, open-source, library	Sì	Windows / Linux	Libreria PyTorch ad alte prestazioni per Gaussian Splatting
Trainer + Visualizzatore	Nerfstudio	trainer, desktop, open-source, framework	Sì	Windows / Linux	Framework modulare per training e sperimentazione
Trainer	OpenSplat	trainer, desktop, open-source	Sì	Windows / Linux	Implementazione nativa in C++ basata su LibTorch
Trainer + Visualizzatore	LichtFeld Studio	trainer, desktop, open-source	Sì	Windows / Linux	Trainer ottimizzato in C++ per Gaussian Splatting

Tabella 3.4: Caratteristiche delle pipeline di training dei trainer open source.

Nome	Input	Formato Output	Dim. Output	Tempo	Livello di accessibilità
Inria gaussian-splatting	Immagini	.ply	Alta	Alto	Avanzato
gsplat	Immagini	.ply	Alta	Medio	Avanzato
Nerfstudio	Immagini / Video / COLMAP	.ply	Bassa	Basso	Avanzato
OpenSplat	Immagini (COLMAP / Nerfstudio)	.ply	Media	Medio	Avanzato
LichtFeld Studio	Immagini / Video / COLMAP	.ply, .spz, .lfs	Alta	Medio	Avanzato

Le implementazioni open source considerate nell’overview mostrano approcci differenti alla progettazione delle pipeline di training, soprattutto in termini di modularità, prestazioni e livello di astrazione dell’interfaccia.

L’implementazione **Inria gaussian-splatting** rappresenta la base di riferimento per molte successive implementazioni. Il sistema offre un controllo diretto sui parametri di training ma richiede una configurazione relativamente complessa della pipeline e un livello di competenza tecnica elevato.

La libreria **gsplat** introduce un approccio differente, fornendo un backend altamente ottimizzato per il rendering e l’ottimizzazione delle gaussiane tramite CUDA e PyTorch.

Un esempio di integrazione di questo backend è rappresentato da **Nerfstudio**, che offre un framework modulare per la gestione dei dataset, la configurazione delle pipeline di training e la visualizzazione dei risultati. In questo contesto la pipeline dedicata al Gaussian Splatting consente di generare modelli relativamente compatti con tempi di training ridotti rispetto ad altre implementazioni.

OpenSplat propone invece un’implementazione basata su C++ e LibTorch, progettata per ridurre la dipendenza da framework Python e facilitare l’integrazione in applicazioni software tradizionali.

Infine, **LichtFeld Studio** adotta un approccio più orientato all’usabilità, offrendo una pipeline completa accompagnata da un’interfaccia grafica che consente di gestire dataset, configurare il training e visualizzare interattivamente le ricostruzioni generate.

3.2.2 Visualizzatori, editor e strumenti di integrazione

Oltre ai trainer, sono stati analizzati diversi strumenti utilizzati per la visualizzazione, l'ispezione, la modifica e l'integrazione dei modelli Gaussian Splatting generati durante il processo di training all'interno di game engine, in particolare **Unity**.

Questa sezione presenta una **overview dei tool considerati** nell'analisi, basata sulla documentazione ufficiale dei progetti e sulle modalità di utilizzo delle diverse pipeline.

Le Tabelle 3.5 e 3.6 riassumono le principali caratteristiche degli strumenti considerati.

Tabella 3.5: Informazioni generali sui visualizzatori, editor e strumenti di integrazione.

Nome	Tag	Piattaforma	Ambiente di esecuzione
SuperSplat	editor, offline, web-based, open-source	Web	Browser
Nerfstudio Viewer	training, streaming, web-based, open-source	Web	Browser
SIBR Viewer	training, offline, desktop, open-source	Desktop	Applicazione standalone
UnityGaussian Splatting	renderer, unity, open-source	Desktop	Unity Engine
Unity-VR-Gaussian-Splatting	xr, renderer, unity, open-source	VR	Unity XR
GaussianSplatting VRViewerUnity	xr, application, unity, open-source	VR	Unity XR

Tabella 3.6: Ruolo e caratteristiche degli strumenti di visualizzazione e integrazione.

Nome	Ruolo	Accesso al modello	Livello di accessibilità	Note
SuperSplat	Editor e visualizzatore	File locali	Facile	Editing e ispezione tramite drag-and-drop nel browser
Nerfstudio Viewer	Visualizzatore training live	Stream della pipeline	Medio	Richiede un processo di training attivo
SIBR Viewer	Monitoraggio training e visualizzazione modelli	Stream o modello locale	Medio	Viewer ufficiale distribuito con il repository GraphDeco
UnityGaussian Splatting	Rendering GS in Unity	Asset locali	Avanzato	Pacchetto renderer per integrazione in progetti Unity
Unity-VR-Gaussian-Splatting	Rendering VR in Unity XR	Asset locali	Avanzato	Estensione VR del renderer Unity-Gaussian-Splatting
GaussianSplatting VRViewerUnity	Visualizzatore VR	Asset locali	Medio	Applicazione VR completa con progetto Unity incluso

Tra gli strumenti considerati nell'overview emergono diverse tipologie che intervengono in fasi differenti della pipeline Gaussian Splatting.

Alcuni strumenti sono progettati principalmente per l'ispezione e la modifica delle ricostruzioni generate. In questa categoria rientra **SuperSplat**, che offre

un ambiente web-based per la visualizzazione e l'editing di modelli Gaussian Splatting tramite caricamento diretto di file locali. Questo tipo di strumenti è particolarmente utile nelle fasi di analisi e post-processing delle ricostruzioni, consentendo di intervenire manualmente sulla struttura dei modelli generati dai trainer.

Un secondo gruppo di strumenti è orientato al monitoraggio del processo di training e alla visualizzazione delle ricostruzioni generate. Il **Nerfstudio Viewer** consente di osservare l'evoluzione della scena durante l'ottimizzazione del modello, ricevendo i dati direttamente dalla pipeline di training tramite streaming.

Il **SIBR Viewer** può essere utilizzato sia per monitorare il training in modalità remota sia come visualizzatore standalone di modelli già addestrati.

Un'ulteriore categoria è costituita dagli strumenti progettati per l'integrazione del rendering Gaussian Splatting all'interno di motori grafici. **UnityGaussianSplatting** fornisce ad esempio un renderer che consente di importare modelli Gaussian Splatting come asset all'interno di progetti Unity, permettendo l'utilizzo delle ricostruzioni in applicazioni interattive.

A partire da questo renderer sono state sviluppate anche soluzioni dedicate alla visualizzazione immersiva. Il progetto **Unity-VR-Gaussian-Splatting** estende il renderer UnityGaussianSplatting introducendo il supporto alla visualizzazione tramite **Unity XR**. In alternativa, **GaussianSplattingVRViewerUnity** propone un visualizzatore VR completo che consente di esplorare modelli Gaussian Splatting senza dover integrare manualmente il renderer all'interno di un progetto Unity.

Nel complesso, gli strumenti analizzati mostrano come la fase di visualizzazione e integrazione delle ricostruzioni Gaussian Splatting possa essere affrontata con approcci differenti, che vanno da strumenti di editing e ispezione delle scene fino a soluzioni progettate per l'integrazione in applicazioni interattive o ambienti immersivi.

3.3 Analisi tecnica delle pipeline di training open source

Questa sezione presenta l'analisi tecnica delle principali implementazioni open source del 3D Gaussian Splatting considerate in questo lavoro. L'obiettivo dell'analisi è valutare il comportamento delle diverse pipeline di training sia dal punto di vista quantitativo, attraverso metriche relative alla dimensione dei modelli generati e al tempo di training, sia dal punto di vista qualitativo, analizzando visivamente le ricostruzioni prodotte.

Le valutazioni sono state condotte utilizzando due scenari distinti, rappresentativi di configurazioni spaziali differenti:

- una scena **indoor**, caratterizzata da uno spazio confinato e da una struttura geometrica relativamente compatta;
- una scena **outdoor**, caratterizzata da un'estensione spaziale maggiore e dalla presenza di elementi ambientali distanti.

Per ciascuno scenario sono state analizzate cinque pipeline di training:

- Inria gaussian-splatting
- gsplat
- OpenSplat
- Nerfstudio
- LichtFeld Studio

Le ricostruzioni generate sono state esportate in formato `.ply` e successivamente ispezionate e post-processate utilizzando l'editor **SuperSplat**. L'analisi considera quindi sia le ricostruzioni **raw**, ottenute direttamente al termine del training, sia le versioni **post-processate**, ottenute tramite operazioni di pulizia della scena.

Le sezioni successive presentano i risultati ottenuti nei due scenari analizzati.

3.4 Analisi tecnica scena indoor

La prima fase dell'analisi è stata condotta su una scena indoor caratterizzata da un ambiente interno con una struttura spaziale relativamente compatta e confini ben definiti. Questo tipo di configurazione consente di osservare con maggiore chiarezza la distribuzione delle gaussiane generate durante il processo di training, poiché la maggior parte della geometria rilevante è concentrata all'interno di un volume limitato.

In tali condizioni, le ricostruzioni prodotte dalle diverse pipeline tendono a generare una struttura centrale ben riconoscibile corrispondente agli elementi principali dell'ambiente, accompagnata da diverse tipologie di artefatti periferici. Tra questi si osservano frequentemente gaussiane isolate, cluster distaccati e strutture allungate che si estendono oltre il volume principale della scena.

L'analisi presentata nelle sezioni successive considera sia le ricostruzioni **raw**, ottenute direttamente al termine del training, sia le versioni **post-processate**.

3.4.1 Benchmark quantitativo

Le Tabelle 3.7 e 3.8 riportano rispettivamente le metriche originali e le metriche normalizzate delle pipeline analizzate. La Figura 3.2 mostra inoltre una rappresentazione grafica delle principali metriche quantitative considerate, consentendo un confronto immediato tra il numero di gaussiane generate, la dimensione dei modelli esportati e il tempo di training delle diverse pipeline.

Tabella 3.7: Metriche originali delle pipeline di training sulla scena indoor.

Tool	# Gaussians	Output Size (MB)	Training Time (min)
Inria gaussian-splatting	955,819	226.1	120
gsplat	1,265,239	284.8	50
OpenSplat	510,870	120.8	60
Nerfstudio	170,150	40.2	30
LichtFeld Studio	1,000,000	236.5	60

Tabella 3.8: Metriche normalizzate rispetto al numero di gaussiane generate.

Tool	Storage / 100k Gaussians (MB)	Training Time / 100k Gaussians (min)
Inria gaussian-splatting	23.7	12.6
gsplat	22.5	4.0
OpenSplat	23.6	11.7
Nerfstudio	23.6	17.6
LichtFeld Studio	23.7	6.0

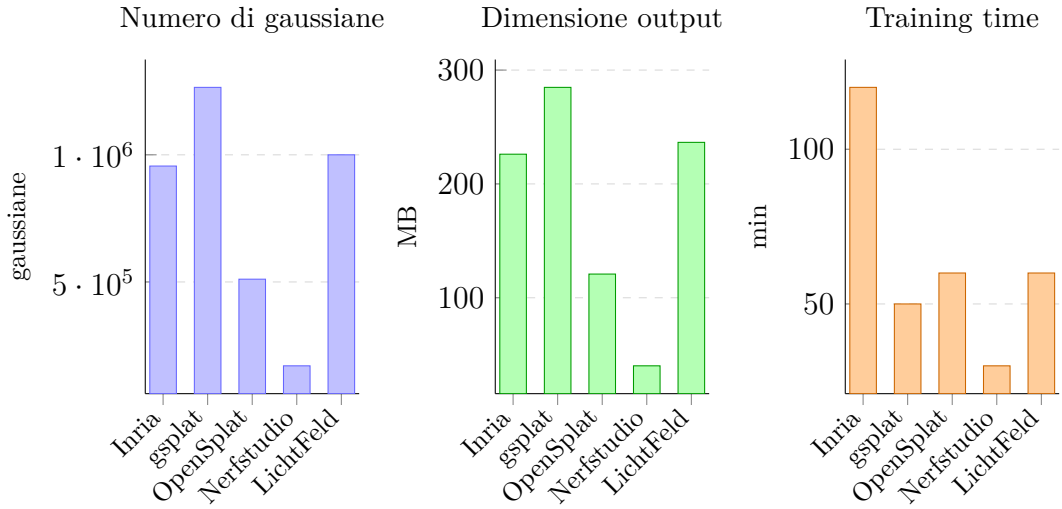


Figura 3.2: Confronto delle principali metriche quantitative delle pipeline di training sulla scena indoor: numero di gaussiane generate, dimensione dei modelli esportati e tempo di training.

Dai risultati emerge come le diverse pipeline producano ricostruzioni con densità e dimensioni significativamente differenti. Come mostrato sia nelle tabelle sia nella rappresentazione grafica della Figura 3.2, alcune pipeline generano modelli più densi e voluminosi, mentre altre producono rappresentazioni più compatte.

La pipeline **gsplat** genera il maggior numero di gaussiane e produce anche il modello con dimensione più elevata. Al contrario, **Nerfstudio** produce la ricostruzione più compatta sia in termini di dimensione del file sia di numero di gaussiane generate.

Per quanto riguarda il tempo di training, l’implementazione originale **Inria gaussian-splatting** risulta la più lenta tra quelle analizzate, mentre **Nerfstudio** mostra il tempo complessivo più ridotto.

Le metriche normalizzate riportate nella Tabella 3.8 consentono di confrontare le pipeline in modo più diretto eliminando l’influenza del diverso numero di gaussiane generate.

La metrica **Storage / 100k Gaussians** rappresenta la quantità di spazio occupata dal modello per ogni 100 000 gaussiane e fornisce una misura dell’efficienza di memorizzazione delle rappresentazioni generate. I valori risultano molto simili tra le diverse pipeline, suggerendo che il costo di memorizzazione per singola gaussiana nei file `.ply` è sostanzialmente comparabile.

La metrica **Training Time / 100k Gaussians** rappresenta invece il tempo necessario per generare un numero equivalente di gaussiane e permette di confrontare l’efficienza computazionale delle pipeline. In questo caso emergono differenze

più marcate: la pipeline **gsplat** mostra il valore più basso, indicando una maggiore efficienza nel processo di training, mentre **Nerfstudio** presenta un tempo normalizzato più elevato, anche a causa del minor numero complessivo di gaussiane generate.

3.4.2 Analisi visiva delle ricostruzioni raw

Le ricostruzioni generate dalle diverse pipeline sono state ispezionate visivamente tramite l'editor SuperSplat al fine di analizzare la distribuzione spaziale delle gaussiane e identificare eventuali artefatti presenti nelle scene.

Le figure seguenti mostrano alcune viste delle ricostruzioni raw generate dai diversi metodi prima dell'applicazione delle operazioni di post-processing.

Inria gaussian-splatting

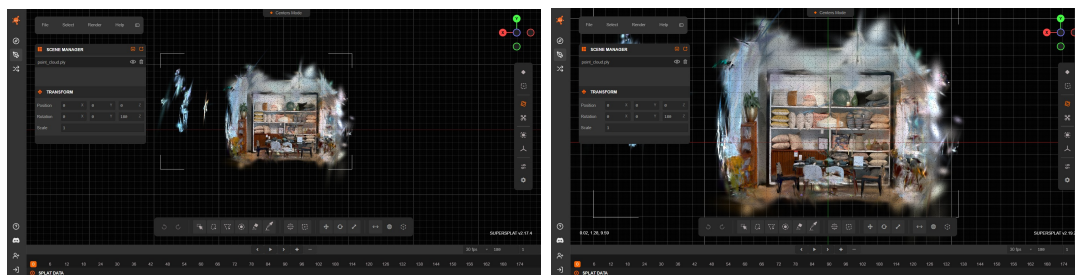


Figura 3.3: Ricostruzione raw della scena indoor generata da Inria gaussian-splatting.

La ricostruzione raw prodotta da **Inria gaussian-splatting** mostra un volume centrale relativamente compatto, con una geometria interna chiaramente riconoscibile. La distribuzione delle gaussiane rimane nel complesso contenuta, anche se sono presenti alcuni outlier periferici e sottili artefatti allungati che si estendono oltre il nucleo principale della scena.

gsplat

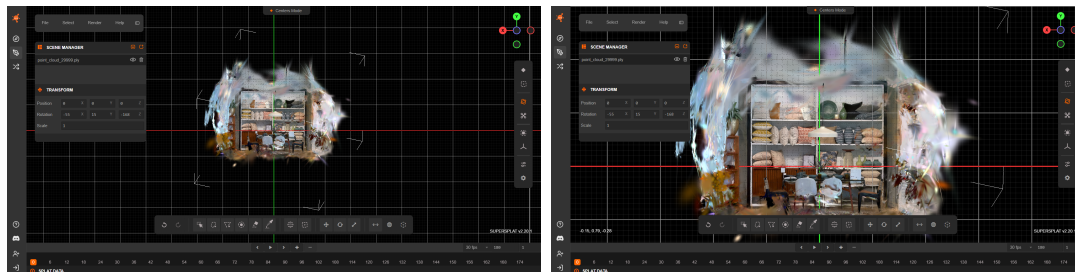


Figura 3.4: Ricostruzione raw della scena indoor generata da gsplat.

La ricostruzione ottenuta con **gsplat** presenta una scena centrale ben definita, circondata da un alone di gaussiane periferiche e da strutture semi-trasparenti distribuite principalmente nelle regioni superiori e laterali. Questi elementi rimangono comunque relativamente vicini alla struttura principale e non formano grandi cluster distaccati.

OpenSplat

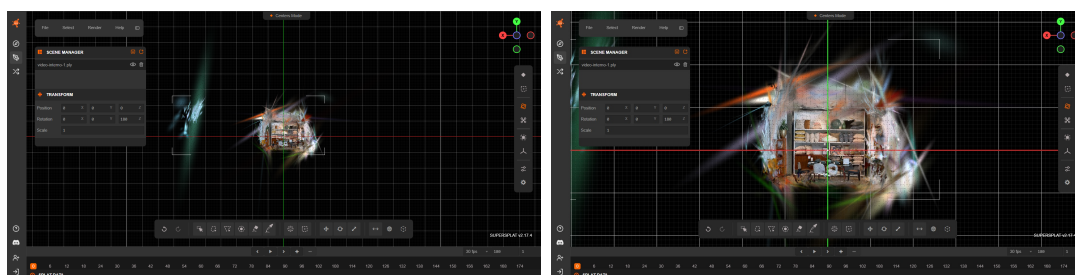


Figura 3.5: Ricostruzione raw della scena indoor generata da OpenSplat.

La ricostruzione raw di **OpenSplat** mostra un nucleo centrale compatto e chiaramente leggibile, ma accompagnato da artefatti allungati e cluster periferici che ampliano l'estensione spaziale del modello attorno alla scena principale.

Nerfstudio

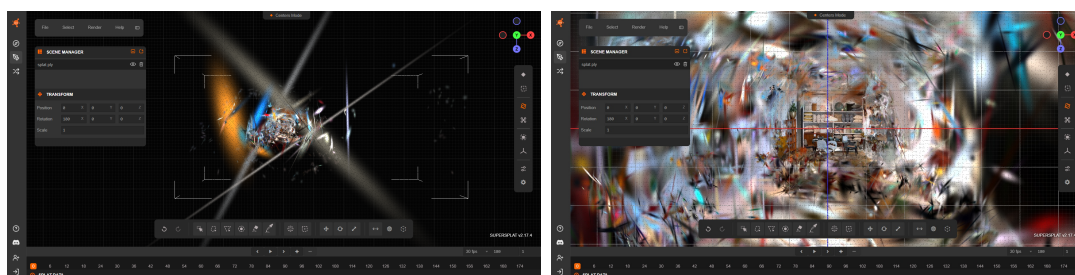


Figura 3.6: Ricostruzione raw della scena indoor generata da Nerfstudio.

La ricostruzione prodotta da **Nerfstudio** presenta un nucleo centrale più sparso rispetto alle altre pipeline, circondato da strutture allungate e cluster periferici isolati. La presenza di gaussiane sottili che si estendono oltre il volume principale riduce la compattezza complessiva della ricostruzione.

LichtFeld Studio

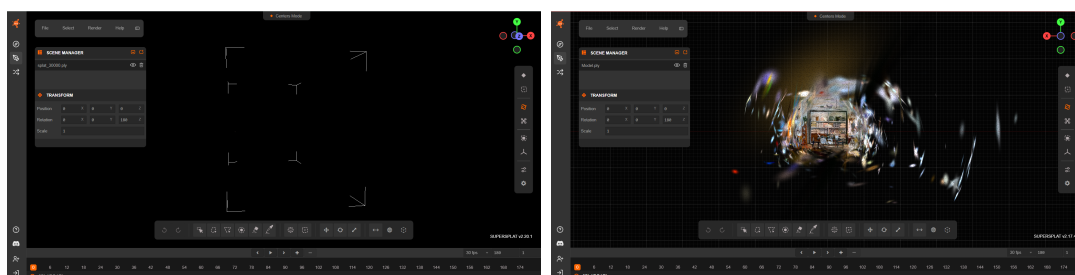


Figura 3.7: Ricostruzione raw della scena indoor generata da LichtFeld Studio.

La ricostruzione raw di **LichtFeld Studio** mostra un nucleo centrale molto denso, circondato da un ampio alone di gaussiane periferiche e da numerosi artefatti distribuiti attorno alla scena. Questo comportamento determina una notevole estensione spaziale del modello rispetto al volume effettivo dell'ambiente indoor.

3.4.3 Impatto delle operazioni di pulizia

Dopo l'ispezione delle ricostruzioni raw, le scene sono state sottoposte a operazioni di pulizia tramite l'editor SuperSplat. Queste operazioni hanno lo scopo di rimuovere gaussiane isolate, cluster distaccati e strutture allungate che non appartengono alla geometria principale della scena.

Nelle ricostruzioni generate dalle diverse pipeline, una parte delle gaussiane prodotte durante il training tende infatti a distribuirsi oltre il volume effettivo dell’ambiente ricostruito. Tali strutture sono generalmente associate a errori di ricostruzione, rumore nei dati di input o a regioni della scena caratterizzate da una copertura visiva limitata.

Le operazioni di pulizia consistono principalmente nella selezione e nella rimozione manuale delle gaussiane periferiche e dei cluster isolati, con l’obiettivo di mantenere solamente le strutture che contribuiscono alla rappresentazione geometrica dell’ambiente.

Per valutare l’impatto di queste operazioni è stata eseguita una nuova analisi quantitativa dei file `.ply` generati dopo il post-processing. In particolare sono stati confrontati il numero di gaussiane presenti nelle ricostruzioni e la dimensione dei file prima e dopo la pulizia della scena.

Tabella 3.9: Impatto delle operazioni di pulizia sul numero di gaussiane della scena indoor.

Tool	Raw Gaussians	Cleaned Gaussians	Δ Gaussians (%)
Inria gaussian-splatting	955,819	518,140	-45.8
gsplat	1,265,239	690,776	-45.4
OpenSplat	510,870	402,531	-21.2
Nerfstudio	170,150	126,841	-25.4
LichtFeld Studio	1,000,000	800,515	-20.0

Tabella 3.10: Impatto delle operazioni di pulizia sulla dimensione dei modelli della scena indoor.

Tool	Raw Size (MB)	Cleaned Size (MB)	Δ Size (%)
Inria gaussian-splatting	226.1	122.6	-45.8
gsplat	284.8	155.5	-45.4
OpenSplat	120.8	95.2	-21.2
Nerfstudio	40.2	30.0	-25.4
LichtFeld Studio	236.5	189.3	-20.0

Le metriche riportate nelle Tabelle 3.9 e 3.10 mostrano come le operazioni di pulizia abbiano un impatto significativo sia sul numero complessivo di gaussiane sia sulla dimensione dei modelli generati.

La riduzione della dimensione dei file `.ply` è direttamente collegata alla rimozione delle gaussiane periferiche che non contribuiscono alla rappresentazione della geometria della scena.

Per facilitare il confronto tra le diverse pipeline, la Figura 3.8 mostra una rappresentazione grafica della riduzione percentuale del numero di gaussiane e della dimensione dei modelli ottenuta dopo le operazioni di pulizia.

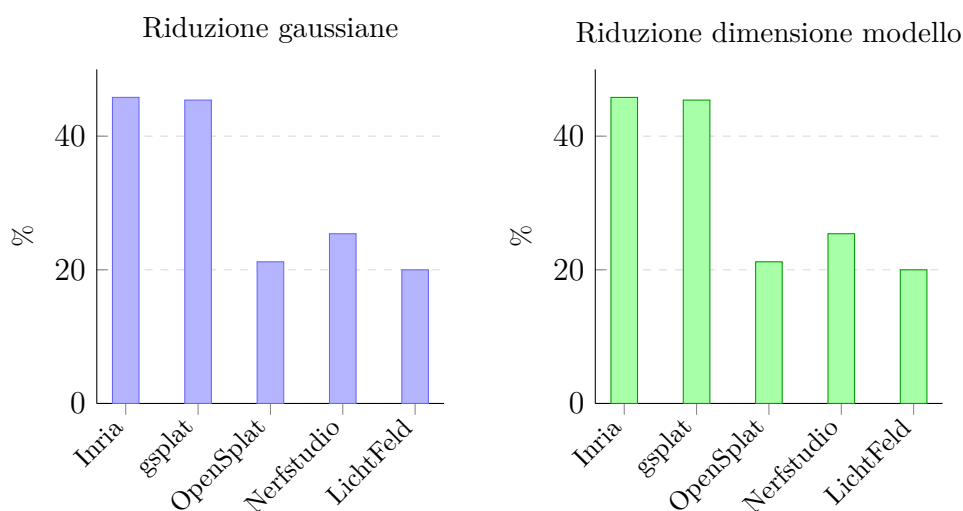


Figura 3.8: Riduzione percentuale del numero di gaussiane e della dimensione dei modelli dopo le operazioni di pulizia nella scena indoor.

3.4.4 Analisi visiva delle ricostruzioni dopo il post-processing

Le ricostruzioni ottenute dopo le operazioni di pulizia sono state analizzate tramite ispezione visiva utilizzando l'editor SuperSplat, al fine di osservare come la rimozione delle gaussiane periferiche influenzi la struttura complessiva della scena.

Le operazioni di post-processing consentono di ridurre l'estensione spaziale delle ricostruzioni e di rendere più leggibile la distribuzione delle gaussiane all'interno del volume della scena.

Le figure seguenti mostrano alcune viste delle ricostruzioni dopo l'applicazione delle operazioni di pulizia.

Inria gaussian-splatting

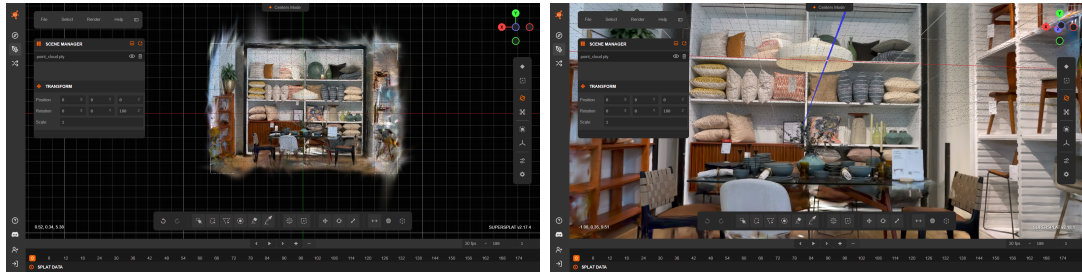


Figura 3.9: Ricostruzione della scena indoor dopo le operazioni di pulizia (Inria gaussian-splatting).

Dopo il post-processing la ricostruzione generata da **Inria gaussian-splatting** risulta più compatta e concentrata attorno alla struttura centrale della scena. La rimozione dei cluster periferici riduce l'estensione spaziale della ricostruzione, migliorando la leggibilità dell'ambiente.

gsplat

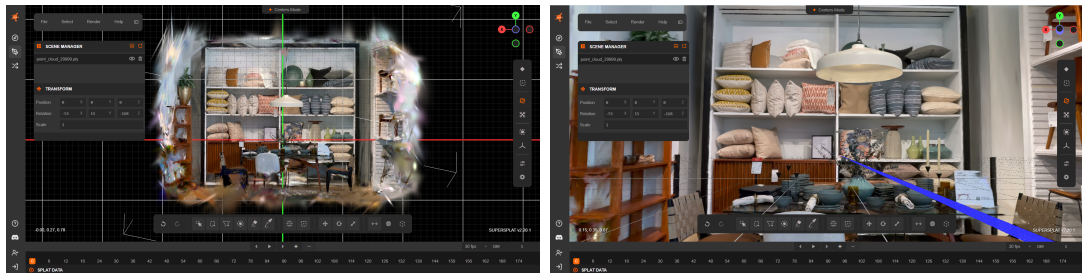


Figura 3.10: Ricostruzione della scena indoor dopo le operazioni di pulizia (gsplat).

Nel caso di **gsplat**, la pulizia della scena elimina gran parte delle gaussiane distribuite nelle regioni superiori e laterali della ricostruzione. La struttura principale dell'ambiente rimane invece ben definita e risulta più chiaramente distinguibile.

OpenSplat

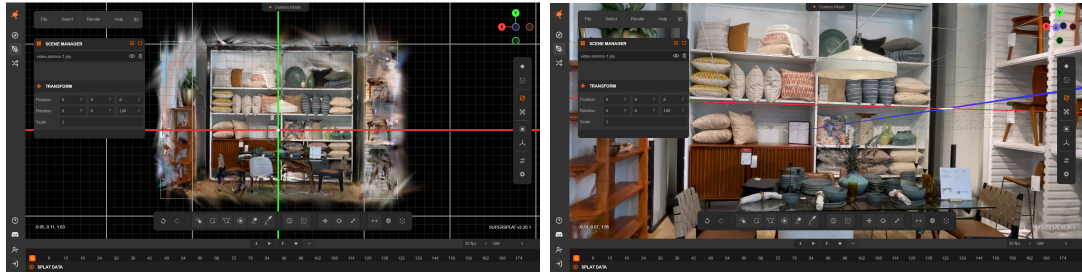


Figura 3.11: Ricostruzione della scena indoor dopo le operazioni di pulizia (OpenSplat).

La ricostruzione ottenuta con **OpenSplat** mostra una significativa riduzione dei cluster periferici dopo la pulizia. La scena principale rimane concentrata nel volume centrale, con una distribuzione delle gaussiane più compatta rispetto alla versione raw.

Nerfstudio

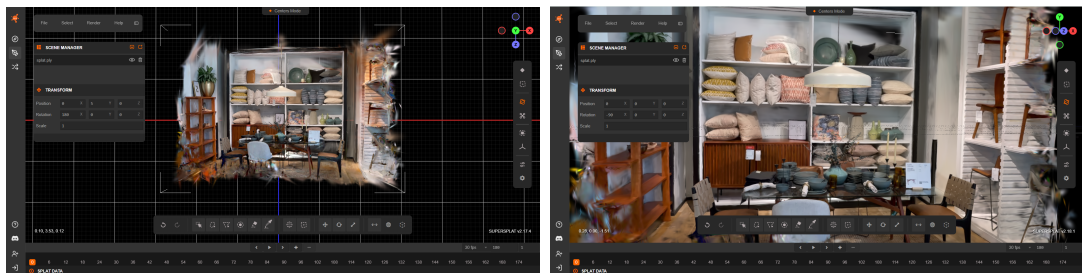


Figura 3.12: Ricostruzione della scena indoor dopo le operazioni di pulizia (Nerfstudio).

Nel caso di **Nerfstudio**, le operazioni di pulizia rimuovono gran parte delle strutture isolate distribuite nello spazio circostante, rendendo la scena più leggibile nonostante la minore densità di gaussiane rispetto ad altre pipeline.

LichtFeld Studio

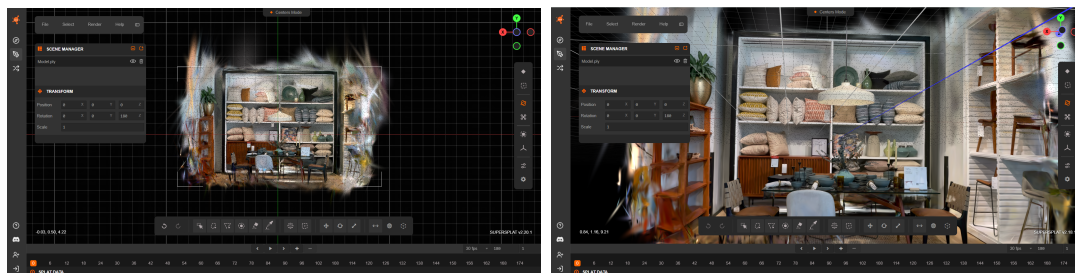


Figura 3.13: Ricostruzione della scena indoor dopo le operazioni di pulizia (LichtFeld Studio).

La ricostruzione prodotta da **LichtFeld Studio** risulta notevolmente più compatta dopo la pulizia. La rimozione delle gaussiane periferiche riduce l'estensione della scena e consente di isolare con maggiore chiarezza la geometria principale dell'ambiente indoor.

3.4.5 Sintesi dei risultati sulla scena indoor

L'analisi della scena indoor evidenzia come le diverse pipeline di training producano ricostruzioni con caratteristiche strutturali e densità significativamente differenti. Alcuni metodi generano modelli molto densi, caratterizzati da un elevato numero di gaussiane e da una maggiore dimensione dei file di output, mentre altri producono ricostruzioni più compatte con un numero ridotto di gaussiane.

L'analisi visiva delle ricostruzioni raw mostra inoltre come tutte le pipeline tendano a generare gaussiane periferiche e cluster isolati al di fuori del volume principale della scena. Le operazioni di post-processing consentono di ridurre significativamente tali artefatti, rendendo le ricostruzioni più compatte e migliorando la leggibilità della struttura geometrica dell'ambiente.

Nel complesso, i risultati ottenuti nello scenario indoor forniscono una prima indicazione del comportamento delle diverse pipeline di training in presenza di scene con geometria relativamente compatta e ben delimitata.

3.5 Analisi tecnica scena outdoor

La seconda fase dell'analisi è stata condotta su una scena outdoor caratterizzata da un ambiente esterno con una maggiore estensione spaziale e dalla presenza di elementi ambientali distanti come vegetazione, cielo e superfici riflettenti.

A differenza delle scene indoor, dove la maggior parte della geometria è contenuta all'interno di un volume limitato, nelle ricostruzioni outdoor le gaussiane possono

distribuirsi su una regione molto più ampia dello spazio. Questo comportamento è dovuto alla presenza di elementi di sfondo e strutture ambientali lontane.

Di conseguenza, le ricostruzioni prodotte dalle diverse pipeline tendono a presentare una struttura centrale associata alla scena principale, accompagnata da numerosi elementi periferici distribuiti nel volume circostante. Tra questi si osservano frequentemente gaussiane isolate, strutture allungate e cluster distaccati corrispondenti a porzioni di vegetazione, cielo o geometria distante.

Come nel caso della scena indoor, l’analisi considera sia le ricostruzioni **raw**, ottenute direttamente al termine del training, sia le versioni **post-processate**.

3.5.1 Benchmark quantitativo

Le Tabelle 3.11 e 3.12 riportano rispettivamente le metriche originali e le metriche normalizzate delle pipeline analizzate. La Figura 3.14 visualizza invece graficamente le principali metriche quantitative considerate, mostrando il numero di gaussiane generate, la dimensione dei modelli esportati e il tempo di training per ciascuna pipeline.

Tabella 3.11: Metriche originali delle pipeline di training sulla scena outdoor.

Tool	# Gaussians	Output Size (MB)	Training Time (min)
Inria gaussian-splatting	777,067	183.8	60
gsplat	1,031,707	232.2	45
OpenSplat	589,291	139.4	50
Nerfstudio	197,545	46.7	25
LichtFeld Studio	1,000,000	236.5	50

Tabella 3.12: Metriche normalizzate rispetto al numero di gaussiane generate.

Tool	Storage / 100k Gaussians (MB)	Training Time / 100k Gaussians (min)
Inria gaussian-splatting	23.7	7.7
gsplat	22.5	4.4
OpenSplat	23.7	8.5
Nerfstudio	23.6	12.7
LichtFeld Studio	23.7	5.0

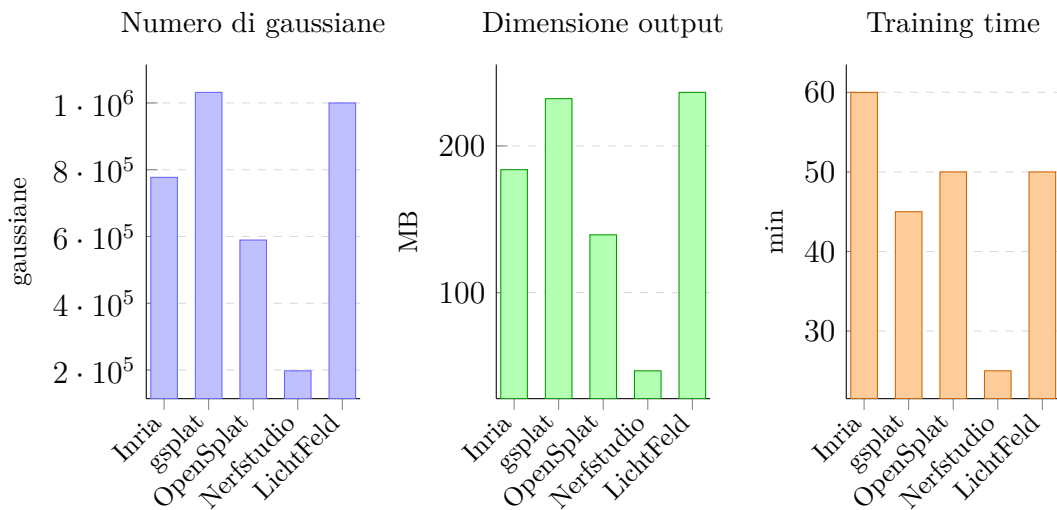


Figura 3.14: Confronto delle principali metriche quantitative delle pipeline di training sulla scena outdoor: numero di gaussiane generate, dimensione dei modelli esportati e tempo di training.

Le pipeline producono ricostruzioni con densità e dimensioni differenti.

La pipeline **gsplat** genera il maggior numero di gaussiane, mentre **Nerfstudio** produce la rappresentazione più compatta in termini di dimensione del modello e numero di gaussiane.

L’implementazione **Inria gaussian-splatting** risulta più lenta rispetto alle altre pipeline, mentre **Nerfstudio** mostra il tempo di training più ridotto.

Le metriche normalizzate permettono di confrontare l’efficienza delle pipeline indipendentemente dal numero totale di gaussiane generate. Come nella scena indoor, i valori di **Storage / 100k Gaussians** risultano molto simili tra i diversi metodi, indicando che il costo di memorizzazione per singola gaussiana è comparabile tra le pipeline.

Al contrario, la metrica **Training Time / 100k Gaussians** evidenzia differenze più marcate: **gsplat** presenta il valore più basso, indicando una maggiore efficienza computazionale, mentre **Nerfstudio** mostra un tempo normalizzato più elevato.

3.5.2 Analisi visiva delle ricostruzioni raw

Le ricostruzioni generate dalle diverse pipeline sono state analizzate tramite ispezione visiva utilizzando l’editor SuperSplat, al fine di valutare la distribuzione spaziale delle gaussiane e la presenza di artefatti nelle scene ricostruite.

Le figure seguenti mostrano alcune viste delle ricostruzioni raw generate dalle diverse pipeline prima dell’applicazione delle operazioni di pulizia.

Inria gaussian-splatting

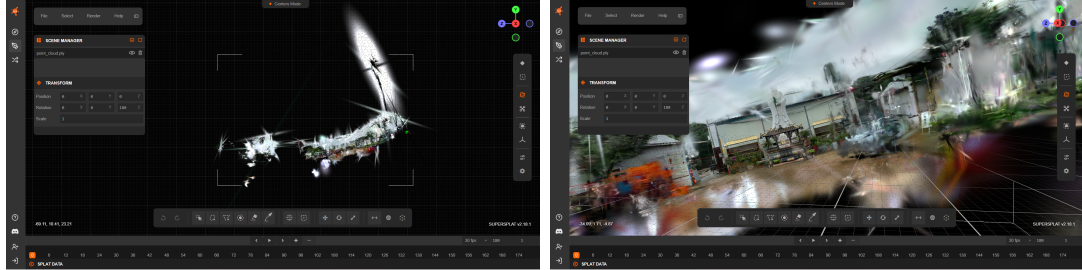


Figura 3.15: Ricostruzione raw della scena outdoor generata da Inria gaussian-splatting.

La ricostruzione raw prodotta da **Inria gaussian-splatting** mostra una scena centrale chiaramente identificabile circondata da gaussiane distribuite in regioni di sfondo associate a vegetazione e contesto ambientale. Sono presenti strutture allungate e gaussiane di grandi dimensioni nelle regioni superiori della scena, associate alle porzioni di cielo.

gsplat

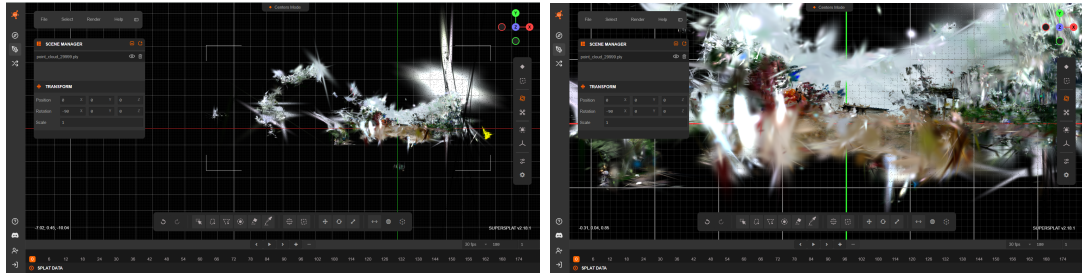


Figura 3.16: Ricostruzione raw della scena outdoor generata da gsplat.

La ricostruzione prodotta da **gsplat** presenta una struttura centrale ben riconoscibile ma accompagnata da numerose gaussiane periferiche distribuite nelle regioni circostanti. Elementi ambientali distanti risultano separati dal volume principale della scena e sono visibili strutture allungate sia vicino alla scena sia nelle regioni più distanti.

OpenSplat

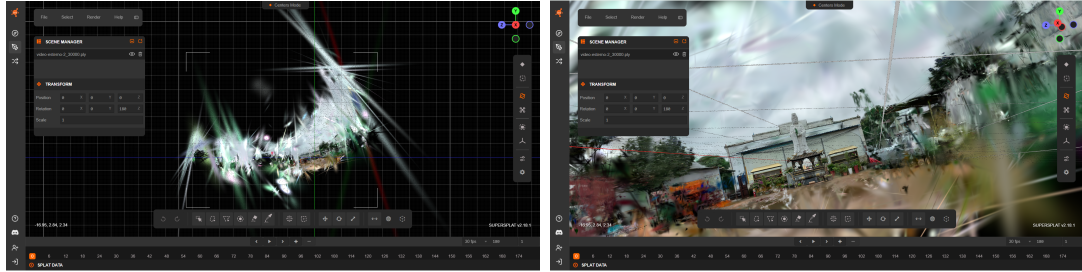


Figura 3.17: Ricostruzione raw della scena outdoor generata da OpenSplat.

La ricostruzione ottenuta con **OpenSplat** mostra una scena centrale riconoscibile ma circondata da un ampio insieme di artefatti allungati che formano un alone irregolare attorno al volume principale della scena. Alcune strutture spike-like si estendono verso le regioni superiori associate al cielo.

Nerfstudio

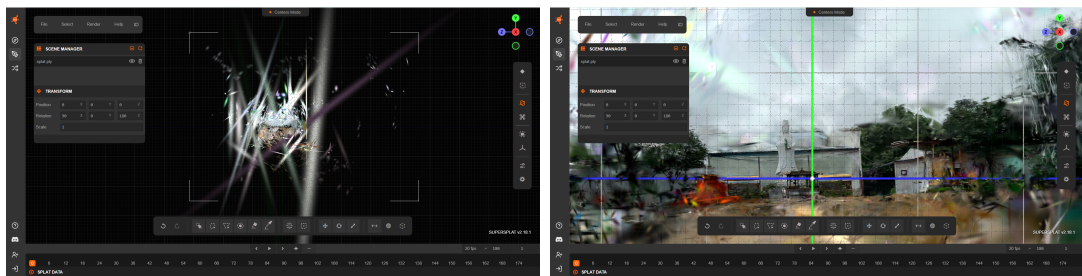


Figura 3.18: Ricostruzione raw della scena outdoor generata da Nerfstudio.

La ricostruzione prodotta da **Nerfstudio** presenta una struttura centrale relativamente compatta circondata da numerosi streak allungati che si estendono nello spazio circostante. Sono inoltre presenti gaussiane isolate distribuite nel volume attorno alla scena.

LichtFeld Studio

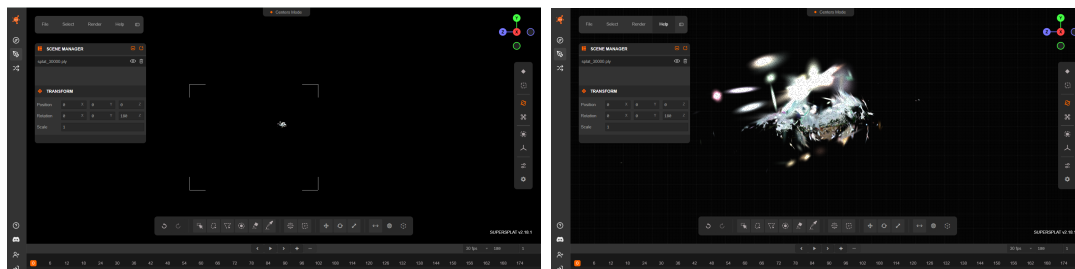


Figura 3.19: Ricostruzione raw della scena outdoor generata da LichtFeld Studio.

La ricostruzione prodotta da **LichtFeld Studio** presenta un nucleo centrale molto denso con la struttura architettonica principale ben riconoscibile. Attorno al nucleo è presente un alone di gaussiane periferiche, mentre alcune strutture allungate sono visibili nelle regioni superiori della scena.

3.5.3 Impatto delle operazioni di pulizia

Dopo l'ispezione delle ricostruzioni raw, le scene outdoor sono state sottoposte a operazioni di pulizia tramite l'editor SuperSplat. A differenza della scena indoor, il processo di pulizia risulta più complesso a causa della presenza di numerosi elementi ambientali distanti — come vegetazione, sfondo urbano o porzioni di cielo — che possono essere legittimamente parte della scena ricostruita.

Per questo motivo le operazioni di pulizia sono state applicate con un approccio conservativo. Il processo ha combinato restrizioni spaziali sul volume della scena, filtri basati sulla distanza, sull'opacità e sulla scala delle gaussiane, insieme a una fase di ispezione manuale finalizzata a distinguere tra artefatti di ricostruzione e geometria effettivamente appartenente all'ambiente.

Per valutare l'impatto di queste operazioni è stata eseguita una nuova analisi quantitativa dei file `.ply` generati dopo il post-processing. In particolare sono stati confrontati il numero di gaussiane presenti nelle ricostruzioni e la dimensione dei modelli prima e dopo la pulizia della scena.

Tabella 3.13: Impatto delle operazioni di pulizia sul numero di gaussiane della scena outdoor.

Tool	Raw Gaussians	Cleaned Gaussians	Δ Gaussians (%)
Inria gaussian-splatting	777,067	369,432	-52.5
gsplat	1,031,707	556,464	-46.1
OpenSplat	589,291	463,802	-21.3
Nerfstudio	197,545	170,923	-13.5
LichtFeld Studio	1,000,000	416,479	-58.4

Tabella 3.14: Impatto delle operazioni di pulizia sulla dimensione dei modelli della scena outdoor.

Tool	Raw Size (MB)	Cleaned Size (MB)	Δ Size (%)
Inria gaussian-splatting	183.8	87.4	-52.4
gsplat	232.2	125.2	-46.1
OpenSplat	139.4	109.7	-21.3
Nerfstudio	46.7	40.4	-13.5
LichtFeld Studio	236.5	98.5	-58.3

Le metriche riportate nelle Tabelle 3.13 e 3.14 mostrano come le operazioni di pulizia abbiano un impatto significativo sia sul numero complessivo di gaussiane sia sulla dimensione dei modelli generati.

La riduzione della dimensione dei file `.ply` è direttamente collegata alla rimozione delle gaussiane periferiche e dei cluster isolati che non contribuiscono alla rappresentazione geometrica dell'ambiente.

Per facilitare il confronto tra le diverse pipeline, la Figura 3.20 mostra una rappresentazione grafica della riduzione percentuale del numero di gaussiane e della dimensione dei modelli ottenuta dopo le operazioni di pulizia.

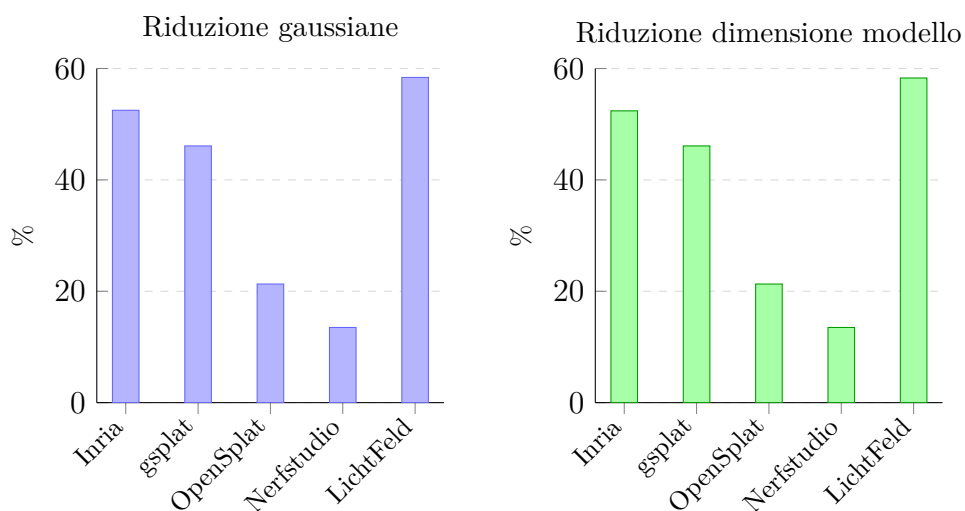


Figura 3.20: Riduzione percentuale del numero di gaussiane e della dimensione dei modelli dopo le operazioni di pulizia nella scena outdoor.

3.5.4 Analisi visiva delle ricostruzioni dopo il post-processing

Le ricostruzioni pulite sono state nuovamente analizzate tramite SuperSplat per osservare l'effetto delle operazioni di post-processing sulla distribuzione delle gaussiane.

Le operazioni di pulizia riducono l'estensione spaziale delle ricostruzioni eliminando cluster periferici e gaussiane isolate, rendendo più evidente la struttura principale della scena.

Le figure seguenti mostrano alcune viste delle ricostruzioni dopo l'applicazione delle operazioni di pulizia.

Inria gaussian-splatting

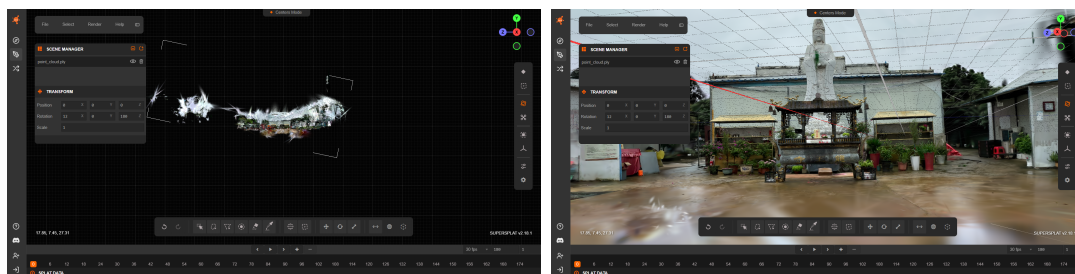


Figura 3.21: Ricostruzione della scena outdoor dopo le operazioni di pulizia (Inria gaussian-splatting).

Dopo il post-processing la ricostruzione prodotta da **Inria gaussian-splatting** mostra una significativa riduzione dell'estensione spaziale della scena. Gran parte delle gaussiane associate alle regioni superiori, in particolare quelle relative al cielo, viene rimossa. La struttura principale dell'ambiente outdoor rimane chiaramente identificabile e stabile. Alcuni artefatti allungati risultano attenuati, sebbene sottili strutture residue siano ancora visibili lungo il perimetro della scena.

gsplat

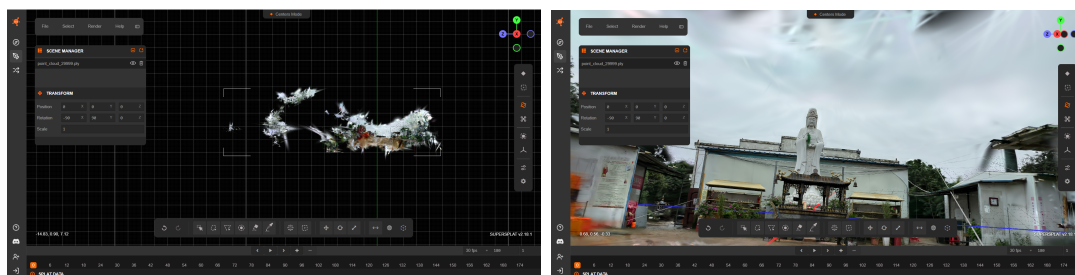


Figura 3.22: Ricostruzione della scena outdoor dopo le operazioni di pulizia (gsplat).

Nel caso di **gsplat**, la pulizia della scena riduce la presenza di gaussiane isolate e di cluster distaccati nelle regioni più lontane della ricostruzione. Alcuni artefatti allungati visibili nella versione raw risultano attenuati, mentre altri persistono nelle regioni superiori associate al cielo. La struttura principale dell'ambiente outdoor appare comunque più chiaramente separata dal rumore circostante, rendendo la scena complessivamente più compatta.

OpenSplat

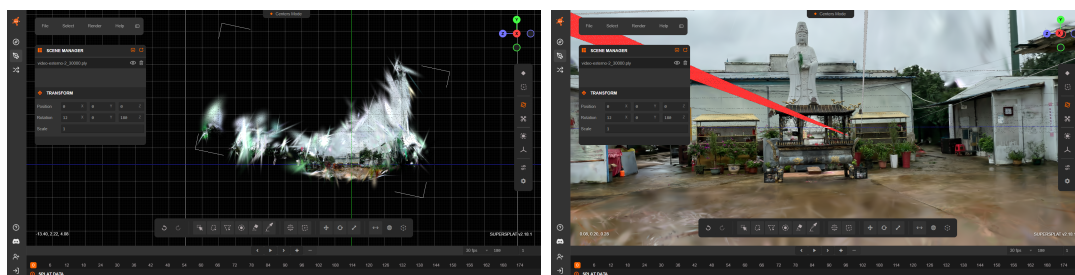


Figura 3.23: Ricostruzione della scena outdoor dopo le operazioni di pulizia (OpenSplat).

La ricostruzione ottenuta con **OpenSplat** mostra una riduzione evidente dell'estensione spaziale dopo le operazioni di pulizia. Molti degli artefatti periferici e

delle strutture allungate vengono rimossi, mentre la geometria centrale della scena rimane ben preservata. Alcune gaussiane associate alle regioni superiori della scena persistono, ma risultano più sottili e localizzate rispetto alla versione raw.

Nerfstudio

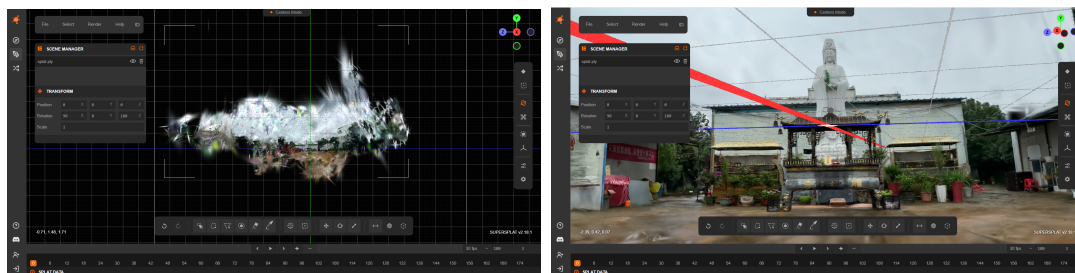


Figura 3.24: Ricostruzione della scena outdoor dopo le operazioni di pulizia (Nerfstudio).

Nel caso di **Nerfstudio**, la pulizia della scena produce una ricostruzione più compatta e concentrata sul volume centrale dell'ambiente. Gran parte dei cluster distanti e degli artefatti allungati presenti nella versione raw viene rimossa. Gli elementi architettonici principali rimangono chiaramente visibili, mentre le gaussiane associate allo sfondo e alla vegetazione risultano significativamente ridotte.

LichtFeld Studio

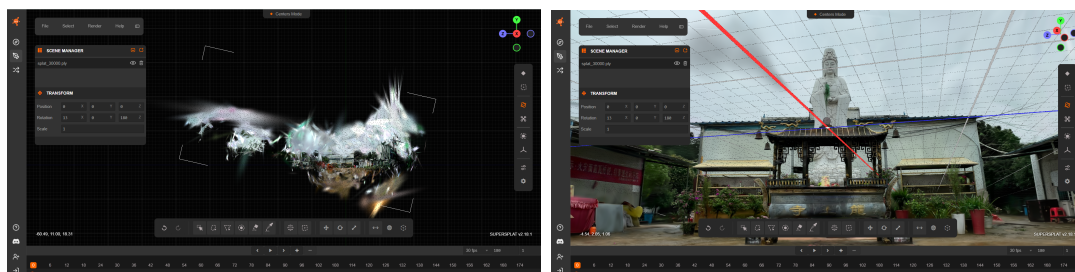


Figura 3.25: Ricostruzione della scena outdoor dopo le operazioni di pulizia (LichtFeld Studio).

La ricostruzione prodotta da **LichtFeld Studio** mostra una scena più centralizzata e meglio delimitata rispetto alla versione raw. La pulizia elimina parte dei cluster fluttuanti e della dispersione periferica di gaussiane, mantenendo comunque la struttura principale dell'ambiente e della vegetazione. Alcuni artefatti sottili e sparsi rimangono lungo i bordi della scena, ma il volume complessivo della ricostruzione appare significativamente più compatto.

3.5.5 Sintesi dei risultati sulla scena outdoor

L'analisi della scena outdoor evidenzia come le pipeline di training generino ricostruzioni caratterizzate da una distribuzione delle gaussiane più ampia rispetto alle scene indoor, a causa della presenza di elementi ambientali distanti.

Le operazioni di pulizia risultano quindi particolarmente importanti per ridurre la presenza di gaussiane periferiche e migliorare la leggibilità della scena, pur mantenendo gli elementi ambientali rilevanti.

Nel complesso, i risultati mostrano come il comportamento delle pipeline vari significativamente in presenza di scene con ampia estensione spaziale, evidenziando differenze nella densità delle ricostruzioni e nella distribuzione delle gaussiane generate.

3.6 Confronto tra scena indoor e scena outdoor

Dopo aver analizzato separatamente i due scenari sperimentali, risulta utile confrontare direttamente il comportamento delle pipeline di training nelle due configurazioni. L'obiettivo di questo confronto è evidenziare come la natura della scena influenzi la distribuzione delle gaussiane, la struttura delle ricostruzioni e l'efficacia delle operazioni di post-processing.

3.6.1 Differenze quantitative nelle ricostruzioni raw

Le Tabelle 3.15 e 3.16 riportano un confronto diretto delle principali metriche quantitative ottenute nei due scenari. La Figura 3.26 mostra una rappresentazione grafica delle differenze osservate nel numero di gaussiane generate e nel tempo di training delle diverse pipeline nei due scenari.

Tabella 3.15: Confronto tra scena indoor e outdoor in termini di numero di gaussiane e dimensione dei modelli.

Tool	Gaussians Indoor	Gaussians Outdoor	Size Indoor (MB)	Size Outdoor (MB)
Inria gaussian-splatting	955,819	777,067	226.1	183.8
gsplat	1,265,239	1,031,707	284.8	232.2
OpenSplat	510,870	589,291	120.8	139.4
Nerfstudio	170,150	197,545	40.2	46.7
LichtFeld Studio	1,000,000	1,000,000	236.5	236.5

Tabella 3.16: Confronto tra scena indoor e outdoor in termini di tempo di training.

Tool	Training Indoor (min)	Training Outdoor (min)
Inria gaussian-splatting	120	60
gsplat	50	45
OpenSplat	60	50
Nerfstudio	30	25
LichtFeld Studio	60	50

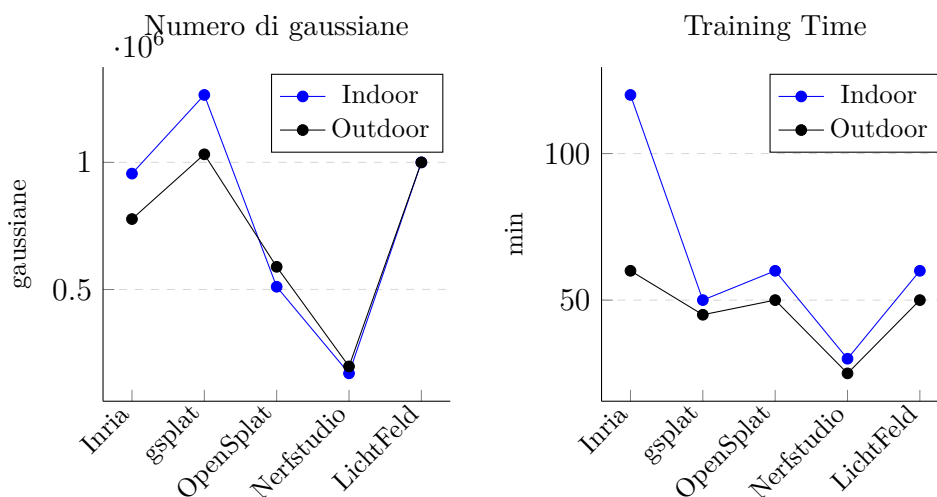


Figura 3.26: Confronto tra scena indoor e outdoor nelle principali metriche di training: numero di gaussiane generate e tempo di training per le pipeline analizzate.

Il confronto quantitativo evidenzia come il comportamento delle pipeline vari tra i due scenari. Alcuni metodi, come **Inria gaussian-splatting** e **gsplat**, producono ricostruzioni più compatte nella scena outdoor, con un numero inferiore di gaussiane e tempi di training ridotti. Al contrario, **OpenSplat** e **Nerfstudio** mostrano un leggero aumento della densità delle ricostruzioni nello scenario outdoor, suggerendo una maggiore ricostruzione degli elementi di sfondo presenti nelle scene esterne. **LichtFeld Studio**, invece, mantiene un numero di gaussiane costante nei due scenari, in quanto la pipeline impone un limite massimo al numero di gaussiane generate durante il training.

3.6.2 Differenze strutturali tra ricostruzioni indoor e outdoor

Oltre alle differenze quantitative, emergono anche differenze strutturali significative tra le ricostruzioni generate nei due ambienti.

Le scene indoor tendono a produrre ricostruzioni con una distribuzione delle gaussiane relativamente compatta. La maggior parte della geometria è contenuta all'interno di un volume limitato e gli artefatti periferici risultano generalmente localizzati nelle vicinanze della struttura principale della scena.

Le scene outdoor presentano invece caratteristiche differenti. Le ricostruzioni raw tendono ad occupare regioni spaziali più ampie, con la presenza di gaussiane associate a elementi ambientali distanti come vegetazione o cielo. In queste condizioni si osservano più frequentemente aloni di gaussiane attorno alla scena principale, strutture allungate che si estendono nello spazio circostante e cluster distaccati associati a porzioni di sfondo.

Di conseguenza, le ricostruzioni outdoor mostrano in generale una distribuzione delle gaussiane più complessa rispetto alle scene indoor, con una maggiore presenza di elementi far-field e regioni periferiche.

3.6.3 Differenze nelle operazioni di pulizia

Le differenze tra i due scenari si riflettono direttamente anche nella complessità delle operazioni di post-processing.

Nel caso delle scene indoor, le operazioni di pulizia risultano generalmente più semplici. Il nucleo della scena è facilmente identificabile e le operazioni di restrizione spaziale possono essere applicate in modo relativamente aggressivo senza compromettere la geometria principale. Di conseguenza, una parte significativa delle gaussiane periferiche può essere rimossa con un rischio limitato di eliminare elementi validi della scena.

Nel caso delle scene outdoor, il processo di pulizia risulta invece più complesso. Le gaussiane distanti possono infatti rappresentare elementi reali della scena, come vegetazione o porzioni di cielo, rendendo più difficile distinguere tra artefatti e geometria legittima. In queste condizioni le operazioni di filtraggio devono essere applicate con maggiore cautela e richiedono spesso una fase più estesa di ispezione manuale.

3.6.4 Impatto quantitativo delle operazioni di pulizia

Le Tabelle 3.17 e 3.18 riportano la variazione percentuale nel numero di gaussiane e nella dimensione dei modelli dopo le operazioni di pulizia. La Figura 3.27 mostra

una rappresentazione grafica della riduzione percentuale del numero di gaussiane nei due scenari.

Tabella 3.17: Riduzione percentuale del numero di gaussiane dopo il post-processing.

Tool	Δ Gaussians Indoor (%)	Δ Gaussians Outdoor (%)
Inria gaussian-splatting	-45.8	-52.5
gsplat	-45.4	-46.1
OpenSplat	-21.2	-21.3
Nerfstudio	-25.4	-13.5
LichtFeld Studio	-20.0	-58.4

Tabella 3.18: Riduzione percentuale della dimensione dei modelli dopo il post-processing.

Tool	Δ Size Indoor (%)	Δ Size Outdoor (%)
Inria gaussian-splatting	-45.8	-52.4
gsplat	-45.4	-46.1
OpenSplat	-21.2	-21.3
Nerfstudio	-25.4	-13.5
LichtFeld Studio	-20.0	-58.3

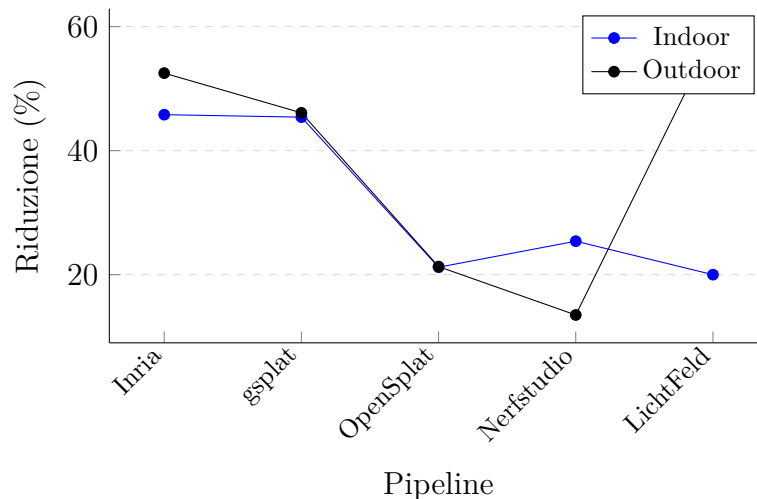


Figura 3.27: Riduzione percentuale del numero di gaussiane dopo il post-processing nelle scene indoor e outdoor.

L'impatto delle operazioni di pulizia varia tra i due scenari. Nelle scene indoor le riduzioni risultano relativamente uniformi tra le pipeline, mentre nelle scene outdoor si osserva una maggiore variabilità.

Nel complesso, il confronto tra i due scenari evidenzia come le ricostruzioni indoor tendano a convergere verso modelli più compatti dopo il post-processing, mentre le ricostruzioni outdoor mantengono strutture più estese e una maggiore presenza di elementi periferici, riflettendo la maggiore complessità geometrica degli ambienti esterni.

3.6.5 Sintesi del confronto tra scenari

Il confronto tra i due scenari evidenzia come la struttura dell'ambiente influenzi la distribuzione delle gaussiane e il comportamento delle pipeline di training. Le scene indoor tendono a generare ricostruzioni più compatte e facilmente interpretabili, mentre le scene outdoor producono distribuzioni spaziali più estese e una maggiore presenza di elementi periferici.

Di conseguenza, le operazioni di post-processing risultano generalmente più semplici negli ambienti indoor, mentre nelle scene outdoor richiedono maggiore attenzione per distinguere tra artefatti e componenti effettive della scena.

3.7 Selezione dello scenario per l'analisi dei visualizzatori XR

Per l'analisi dei visualizzatori XR è stata selezionata la scena **indoor** utilizzata nelle sezioni precedenti per il confronto tra le pipeline di training.

Come evidenziato nell'analisi comparativa tra scenari indoor e outdoor, le ricostruzioni indoor presentano una distribuzione delle gaussiane più compatta e una struttura spaziale più facilmente delimitabile. Questo rende più semplici e controllabili le operazioni di pulizia delle ricostruzioni, permettendo di isolare con maggiore chiarezza la geometria principale della scena.

Al contrario, nelle scene outdoor la presenza di elementi ambientali distanti produce una distribuzione delle gaussiane più estesa e rende le operazioni di post-processing più complesse.

Per questi motivi, la valutazione dei visualizzatori XR è stata condotta utilizzando le ricostruzioni **post-processate della scena indoor**, generate dalle cinque pipeline di training analizzate in precedenza.

3.8 Analisi tecnica dei visualizzatori XR

Questa sezione presenta l'analisi sperimentale degli strumenti utilizzati per la visualizzazione immersiva delle ricostruzioni basate su 3D Gaussian Splatting. L'obiettivo dell'analisi è valutare il comportamento dei diversi strumenti di visualizzazione XR sia dal punto di vista qualitativo, osservando la stabilità delle ricostruzioni durante l'interazione immersiva, sia dal punto di vista prestazionale, analizzando le caratteristiche di rendering in tempo reale.

Le valutazioni sono state condotte utilizzando le ricostruzioni della **scena indoor** analizzata nelle sezioni precedenti.

Sono stati considerati due strumenti basati su **Unity** per la visualizzazione XR delle ricostruzioni Gaussian Splatting:

- un **tool di integrazione per Unity XR**, progettato per consentire l'utilizzo di modelli Gaussian Splatting all'interno di applicazioni immersive sviluppate con Unity;
- un **visualizzatore XR**, sviluppato per il caricamento e l'ispezione diretta delle ricostruzioni Gaussian Splatting in ambiente immersivo.

Per valutare il comportamento dei due strumenti è stata condotta una fase di **ispezione visiva qualitativa** delle ricostruzioni durante l'interazione immersiva. La valutazione è stata effettuata utilizzando una scala ordinale a cinque livelli (1–5), dove il valore 1 indica la presenza di artefatti visivi gravi e chiaramente percepibili, mentre il valore 5 indica una visualizzazione stabile e visivamente coerente.

L'analisi qualitativa è stata condotta considerando i seguenti criteri di valutazione:

- **Surface Solidity**, che misura la continuità geometrica delle superfici e l'assenza di buchi o frammentazioni nella scena;
- **Motion Parallax Stability**, che valuta la coerenza della percezione della profondità durante il movimento della testa;
- **Edge Stability**, che analizza la stabilità dei contorni e l'eventuale presenza di fenomeni di *flickering* o *shimmering*;
- **Layering / Transparency Behavior**, che osserva il comportamento delle superfici sovrapposte e la corretta gestione della trasparenza e dell'occlusione;
- **Near-Object Robustness**, che valuta la stabilità della visualizzazione durante l'ispezione ravvicinata degli oggetti della scena.

Le sezioni successive presentano l’analisi qualitativa del comportamento dei due strumenti durante la visualizzazione immersiva delle ricostruzioni e, dopo aver selezionato il visualizzatore da utilizzare, il confronto delle prestazioni di rendering dei modelli generati dalle diverse pipeline di training osservate durante l’esecuzione.

3.8.1 Analisi visiva delle ricostruzioni nei visualizzatori XR

Le ricostruzioni della scena indoor sono state analizzate tramite analisi visiva nei due strumenti XR considerati, al fine di valutare il comportamento delle ricostruzioni durante la visualizzazione immersiva. L’analisi è stata condotta utilizzando le versioni **post-processate** dei modelli, in modo da garantire condizioni di valutazione coerenti tra i diversi visualizzatori.

Le sottosezioni successive presentano i risultati dell’analisi qualitativa osservata nei due strumenti XR.

Unity-VR-Gaussian-Splatting (ninjamode)

Il primo strumento analizzato è **Unity-VR-Gaussian-Splatting (ninjamode)**, che costituisce un tool di integrazione per Unity progettato per consentire l’utilizzo di ricostruzioni Gaussian Splatting all’interno di applicazioni immersive sviluppate con il motore Unity.

La Tabella 3.19 riporta i punteggi assegnati alle diverse ricostruzioni sulla base dei criteri di valutazione descritti in precedenza.

Tabella 3.19: Valutazione qualitativa delle ricostruzioni nel visualizzatore Unity-VR-Gaussian-Splatting (ninjamode).

Trainer	Surfaces	Parallax	Edges	Layering	Near Objects
Inria gaussian-splatting	5	5	4	4	4
gsplat	4	5	4	4	4
OpenSplat	4	5	4	3	5
Nerfstudio	2	4	3	2	3
LichtFeld Studio	3	2	2	2	3

Nel complesso, la ricostruzione generata da **Inria gaussian-splatting** mostra superfici solide e strutturalmente coerenti, con solo occasionali artefatti spike-like osservabili durante la rotazione della testa.

La ricostruzione ottenuta con **gsplat** presenta un comportamento simile, ma con superfici leggermente più fluide e la comparsa di artefatti spike-like durante il movimento.

Nel caso di **OpenSplat**, le superfici risultano moderatamente fluide e alcune gaussiane colorate posizionate dietro le pareti possono diventare visibili durante il movimento, indicando una certa instabilità nella gestione dei livelli di profondità.

La ricostruzione generata da **Nerfstudio** mostra invece discontinuità strutturali più evidenti, con buchi nelle pareti e artefatti di trasparenza nel pavimento.

Infine, la ricostruzione ottenuta con **LichtFeld Studio** presenta una maggiore instabilità durante il movimento, con vibrazioni diffuse della scena e la presenza di gaussiane proiettate sulle superfici di sfondo.

GaussianSplattingVRViewerUnity (clarte53)

Il secondo strumento analizzato è **GaussianSplattingVRViewerUnity (clarte53)**, che si configura come un visualizzatore XR sviluppato per il caricamento e la visualizzazione diretta di ricostruzioni Gaussian Splatting in ambiente immersivo.

La Tabella 3.20 riporta i punteggi assegnati alle diverse ricostruzioni sulla base dei criteri di valutazione definiti in precedenza.

Tabella 3.20: Valutazione qualitativa delle ricostruzioni nel visualizzatore GaussianSplattingVRViewerUnity (clarte53).

Trainer	Surfaces	Parallax	Edges	Layering	Near Objects
Inria gaussian-splatting	5	3	3	4	4
gsplat	4	3	3	3	4
OpenSplat	4	3	3	3	4
Nerfstudio	1	4	2	1	3
LichtFeld Studio	3	1	4	3	3

Nel caso della ricostruzione generata con **Inria gaussian-splatting**, le superfici risultano generalmente solide e ben definite, ma durante il movimento della testa si osservano fenomeni di rallentamento della scena e un effetto di trascinamento che può influenzare la stabilità della parallasse.

La ricostruzione ottenuta con **gsplat** presenta superfici complessivamente stabili, ma durante il movimento possono comparire leggeri effetti di trasparenza nelle pareti e una certa instabilità nella percezione della profondità.

Nel caso di **OpenSplat**, le superfici risultano moderatamente fluide e durante il movimento possono emergere artefatti di trasparenza simili a quelli osservati nelle ricostruzioni generate con **gsplat**.

La ricostruzione generata con **Nerfstudio** mostra invece discontinuità strutturali più marcate, con pareti perforate, buchi nella geometria e artefatti di trasparenza nel pavimento.

Infine, la ricostruzione ottenuta con **LichtFeld Studio** presenta una qualità visiva statica relativamente elevata, ma il comportamento durante il movimento

risulta fortemente compromesso da rallentamenti significativi della scena e da evidenti effetti di trascinamento durante l'interazione immersiva.

3.8.2 Confronto tra i visualizzatori XR

Il confronto diretto tra i due strumenti consente di evidenziare alcune differenze nel comportamento della visualizzazione immersiva delle ricostruzioni Gaussian Splatting.

Per quanto riguarda la **solidità strutturale delle superfici**, entrambi i visualizzatori preservano una buona coerenza geometrica nelle ricostruzioni generate con **Inria gaussian-splatting**. Tuttavia, nel caso di **Unity-VR-Gaussian-Splatting (ninjamode)** le superfici appaiono generalmente più stabili durante il movimento, mentre nel visualizzatore **GaussianSplattingVRViewerUnity (clarte53)** la scena tende a mostrare fenomeni di rallentamento e trascinamento durante l'interazione.

Per le ricostruzioni generate con **gsplat** e **OpenSplat**, la solidità geometrica rimane complessivamente accettabile in entrambi gli strumenti. Nel caso di **ninjamode** le instabilità risultano principalmente associate a una lieve fluidità delle superfici o a problemi di stratificazione delle gaussiane, mentre in **clarte53** le instabilità risultano maggiormente legate a rallentamenti della scena, effetti di trascinamento e incoerenze nella parallasse durante il movimento.

Nel caso delle ricostruzioni generate con **Nerfstudio**, entrambi i visualizzatori evidenziano chiaramente le debolezze strutturali della ricostruzione, con la presenza di buchi nelle pareti e artefatti di trasparenza.

Per le ricostruzioni ottenute con **LichtFeld Studio**, entrambi i visualizzatori mostrano comportamenti instabili, ma con caratteristiche differenti. Nel caso di **ninjamode** si osservano vibrazioni diffuse e problemi di ordinamento delle gaussiane, mentre nel caso di **clarte53** il comportamento è dominato da rallentamenti significativi della scena e da un evidente effetto di trascinamento durante il movimento.

Nel complesso, **Unity-VR-Gaussian-Splatting (ninjamode)** mostra un comportamento più stabile durante il movimento e garantisce una maggiore coerenza della visualizzazione immersiva. Al contrario, **GaussianSplattingVRViewerUnity (clarte53)** presenta più frequentemente fenomeni di rallentamento, trascinamento e instabilità della parallasse, che riducono la qualità complessiva dell'esperienza immersiva.

3.8.3 Scelta del visualizzatore per l'analisi delle prestazioni

Sulla base delle osservazioni emerse dal confronto qualitativo tra i due strumenti, **Unity-VR-Gaussian-Splatting (ninjamode)** è stato selezionato come visualizzatore di riferimento per le successive analisi prestazionali.

Come evidenziato nella sezione precedente, ninjamode mostra nel complesso un comportamento più stabile durante la visualizzazione immersiva, con una maggiore coerenza della scena durante il movimento e una minore presenza di fenomeni di rallentamento o trascinamento dell’immagine rispetto al visualizzatore **GaussianSplattingVRViewerUnity (clarte53)**.

Inoltre, ninjamode risulta particolarmente rilevante ai fini di questo lavoro in quanto costituisce un **tool di integrazione per Unity XR**. Questo lo rende uno strumento più adatto all’integrazione delle ricostruzioni Gaussian Splatting all’interno di applicazioni immersive sviluppate con il motore Unity, scenario che rappresenta il contesto di utilizzo principale considerato in questa tesi.

Per questi motivi, le analisi quantitative delle prestazioni di rendering sono state condotte utilizzando ninjamode come visualizzatore di riferimento.

3.8.4 Analisi delle prestazioni dei modelli nel visualizzatore XR

Dopo aver selezionato **Unity-VR-Gaussian-Splatting (ninjamode)** come visualizzatore di riferimento, è stata condotta un’analisi quantitativa delle prestazioni di rendering delle ricostruzioni generate dalle diverse pipeline di training.

Le misurazioni sono state effettuate registrando le prestazioni di rendering durante l’esecuzione delle ricostruzioni all’interno del visualizzatore. I valori riportati nella Tabella 3.21 corrispondono alla media di due acquisizioni indipendenti. Per ogni scena è stato considerato un intervallo di acquisizione di 30 secondi successivo alla fase di caricamento iniziale.

Tabella 3.21: Prestazioni di rendering delle ricostruzioni nel visualizzatore Unity-VR-Gaussian-Splatting (ninjamode).

Trainer	Avg FPS	Avg Frame Time (ms)	Avg Max Frame Time (ms)
Inria gaussian-splatting	34.5	29.00	52.92
gsplat	32.0	31.30	56.88
OpenSplat	37.8	26.62	51.43
Nerfstudio	65.0	15.39	30.89
LichtFeld Studio	17.2	58.35	82.62

La Figura 3.28 mostra il confronto tra le diverse pipeline in termini di frame rate medio, mentre la Figura 3.29 riporta il confronto tra i tempi di rendering dei frame.

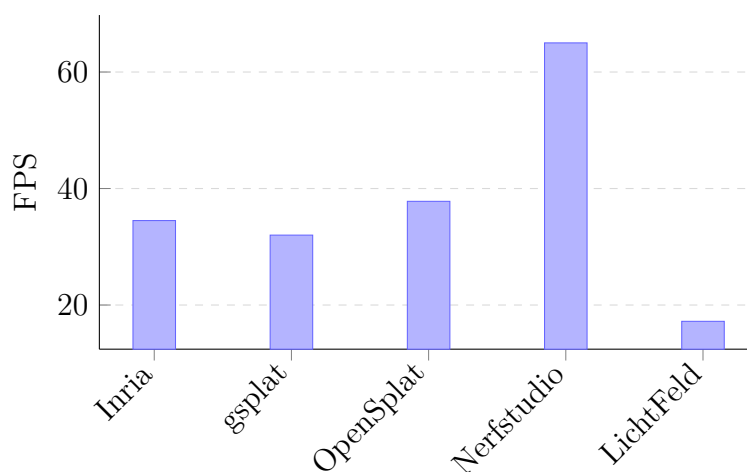


Figura 3.28: Frame rate medio delle ricostruzioni nel visualizzatore Unity-VR-Gaussian-Splatting (ninjamode).

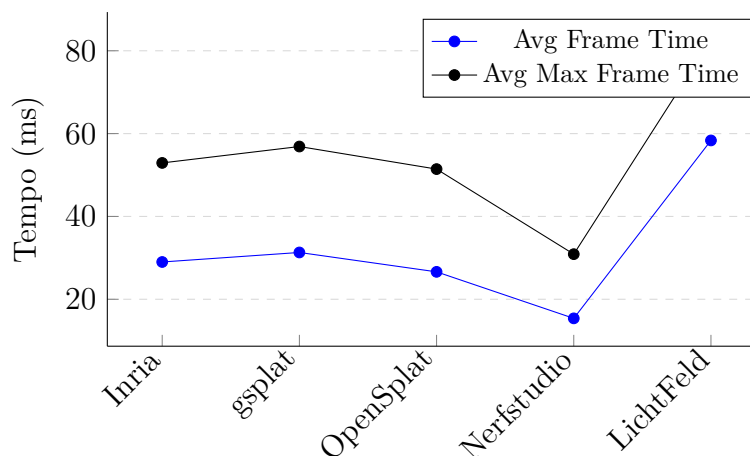


Figura 3.29: Tempo medio di rendering dei frame e tempo massimo osservato durante l'esecuzione nel visualizzatore Unity-VR-Gaussian-Splatting (ninjamode).

Dai risultati emerge come le ricostruzioni generate dalle diverse pipeline di training presentino comportamenti significativamente differenti in termini di prestazioni di rendering.

La ricostruzione generata con **Nerfstudio** mostra il frame rate medio più elevato e il tempo medio di rendering più basso, indicando le migliori prestazioni in tempo reale tra le pipeline di training analizzate.

Al contrario, **LichtFeld Studio** presenta il frame rate medio più basso e i tempi di rendering più elevati, indicando le prestazioni meno favorevoli durante la

visualizzazione immersiva.

La ricostruzione generata con **OpenSplat** mostra un buon equilibrio tra qualità visiva e prestazioni di rendering, con valori di frame rate superiori rispetto a quelli osservati per **Inria gaussian-splatting** e **gsplat**. Queste ultime due pipeline mostrano invece prestazioni complessivamente simili tra loro, con valori di frame rate intermedi e tempi di rendering comparabili.

3.8.5 Sintesi dei risultati dell'analisi dei visualizzatori XR

L'analisi condotta sui visualizzatori XR evidenzia come il comportamento delle ricostruzioni Gaussian Splatting durante la visualizzazione immersiva dipenda sia dalla qualità delle ricostruzioni generate dalle pipeline di training sia dalle caratteristiche del visualizzatore utilizzato.

Dal confronto tra i due strumenti analizzati emerge come **Unity-VR-Gaussian-Splatting (ninjamode)** offra nel complesso una maggiore stabilità durante la visualizzazione immersiva, garantendo una gestione del movimento e della parallasse più coerente e un'esperienza complessivamente più stabile. Al contrario, **Gaussian-SplattingVRViewerUnity (clarte53)** mostra più frequentemente fenomeni di rallentamento della scena, effetti di trascinamento e instabilità durante il movimento dell'utente, che possono ridurre la qualità complessiva dell'esperienza immersiva.

L'analisi quantitativa delle prestazioni nel visualizzatore ninjamode mostra inoltre differenze significative tra le ricostruzioni generate dalle diverse pipeline di training. In particolare, **Nerfstudio** raggiunge il frame rate medio più elevato e il tempo di rendering più basso, indicando le migliori prestazioni in tempo reale tra le pipeline analizzate. Al contrario, **LichtFeld Studio** presenta i valori di frame rate più bassi e i tempi di rendering più elevati, risultando la soluzione meno efficiente dal punto di vista delle prestazioni.

La ricostruzione generata con **OpenSplat** mostra prestazioni di rendering relativamente elevate, mentre **Inria gaussian-splatting** e **gsplat** presentano valori intermedi e complessivamente comparabili.

Nel complesso, i risultati mostrano come la qualità dell'esperienza XR dipenda non solo dal visualizzatore utilizzato ma anche dalle caratteristiche delle ricostruzioni generate dalle pipeline di training. La combinazione tra ricostruzione strutturalmente stabile e visualizzatore capace di gestire correttamente il rendering delle gaussiane risulta quindi fondamentale per ottenere una visualizzazione immersiva efficace delle scene ricostruite.

3.9 Selezione degli strumenti per ambienti immersivi

Le analisi condotte nelle sezioni precedenti hanno permesso di confrontare sia le pipeline di training per 3D Gaussian Splatting sia gli strumenti disponibili per la visualizzazione immersiva delle ricostruzioni.

Dal confronto tra i visualizzatori XR è emerso come **Unity-VR-Gaussian-Splatting (ninjamode)** mostri nel complesso un comportamento più stabile durante la visualizzazione immersiva rispetto a **GaussianSplattingVRViewerUnity (clarte53)**. Inoltre, essendo progettato come tool di integrazione per Unity XR, risulta particolarmente adatto all'utilizzo delle ricostruzioni Gaussian Splatting all'interno di applicazioni immersive basate su Unity.

Per quanto riguarda le pipeline di training, i risultati qualitativi e quantitativi mostrano come **Inria gaussian-splatting** rappresenti la soluzione più equilibrata in termini di qualità delle ricostruzioni e stabilità durante la visualizzazione immersiva.

Sulla base di queste osservazioni, **ninjamode** è stato selezionato come visualizzatore XR di riferimento, mentre **Inria gaussian-splatting** è stato scelto come trainer per la generazione delle ricostruzioni utilizzate negli ambienti immersivi.

Capitolo 4

Ambiente immersivo e sperimentazione sugli utenti

4.1 Descrizione generale dell'esperimento

Lo studio sperimentale ha l'obiettivo di valutare la Quality of Experience (QoE) delle ricostruzioni generate tramite 3D Gaussian Splatting quando visualizzate in un ambiente immersivo mediante visore di realtà virtuale.

L'analisi si concentra su tre dimensioni principali dell'esperienza utente: la qualità visiva percepita della ricostruzione, il senso di presenza nell'ambiente virtuale e il comfort dell'esperienza immersiva.

L'esperimento è stato progettato utilizzando un design **within-subject**, nel quale ogni partecipante sperimenta entrambe le condizioni sperimentali. Per ridurre possibili effetti di ordine nella presentazione degli stimoli, la sequenza delle condizioni è stata alternata tra i partecipanti.

4.1.1 Ambiente immersivo

L'esperienza immersiva ha inizio in una lobby virtuale progettata come spazio introduttivo dell'esperimento. Questa fase consente ai partecipanti di familiarizzare con il visore e con la modalità di osservazione delle ricostruzioni tridimensionali.

L'ambiente della lobby utilizzato per la fase di familiarizzazione è mostrato nella Figura 4.1.

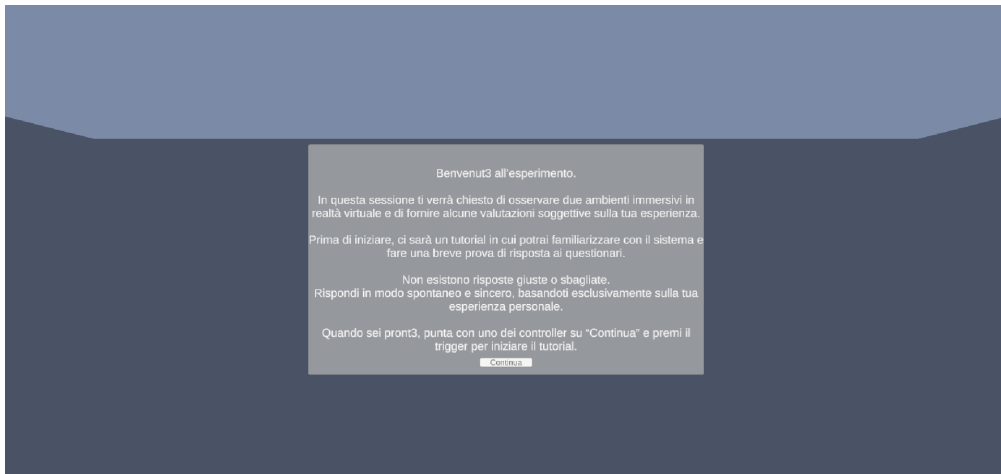


Figura 4.1: Lobby iniziale dell'esperimento VR utilizzata per la fase di familiarizzazione dei partecipanti.

Durante la fase di familiarizzazione viene mostrato un video orbit della ricostruzione *Garden*, presentata nel lavoro originale sul 3D Gaussian Splatting [7]. Questo contenuto è utilizzato come riferimento visivo dello stato dell'arte della tecnica prima dell'inizio delle condizioni sperimentali.

4.1.2 Condizioni sperimentali

L'esperimento prevede due condizioni sperimentali nelle quali la ricostruzione osservata nel visore viene confrontata con differenti riferimenti della scena reale.

Entrambe le condizioni sono progettate per valutare la capacità delle ricostruzioni generate tramite 3D Gaussian Splatting di rappresentare in modo plausibile un ambiente fisico reale quando osservato in realtà virtuale. La differenza tra le due configurazioni riguarda il tipo di riferimento percettivo utilizzato per il confronto.

Ambiente A

In questa condizione i partecipanti osservano inizialmente un video della scena reale utilizzata per generare la ricostruzione. Successivamente visualizzano la stessa scena ricostruita tramite 3D Gaussian Splatting all'interno dell'ambiente immersivo.

La ricostruzione della stanza utilizzata in questa condizione sperimentale è illustrata nella Figura 4.2.

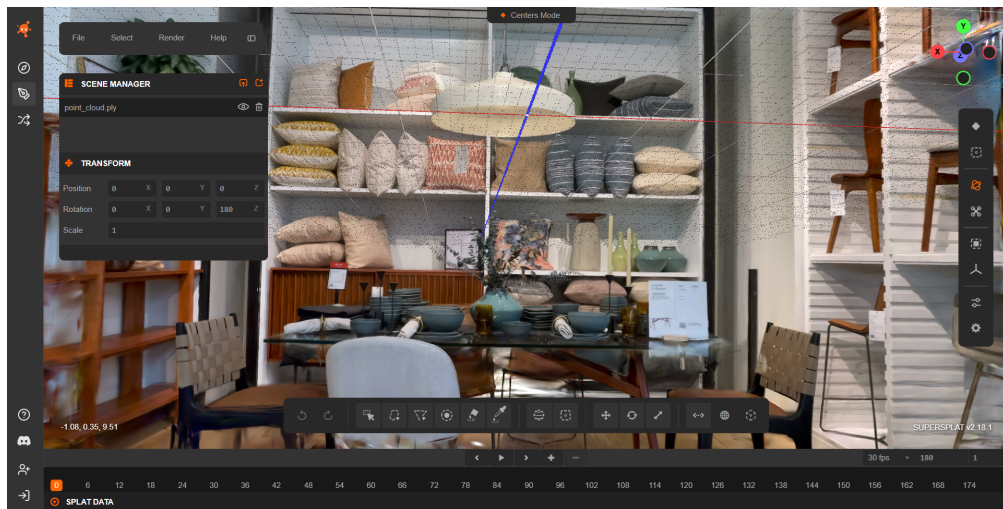


Figura 4.2: Ricostruzione della scena indoor utilizzata nella condizione sperimentale di confronto con il video della scena reale.

Questa configurazione consente di valutare la qualità visiva percepita della ricostruzione confrontandola con un riferimento video della scena reale osservato immediatamente prima dell'esperienza immersiva.

Ambiente B

Nella seconda condizione la ricostruzione osservata nel visore viene confrontata direttamente con l'ambiente fisico reale in cui si svolge l'esperimento.

Prima di indossare il visore i partecipanti osservano direttamente la stanza reale per alcuni minuti, costruendo un riferimento percettivo della scena. Successivamente visualizzano la ricostruzione immersiva della stessa stanza.

La ricostruzione della stanza utilizzata in questa condizione sperimentale è illustrata nella Figura 4.3.

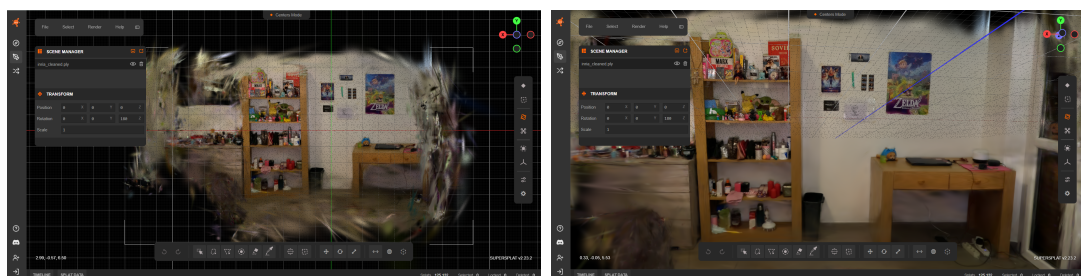


Figura 4.3: Ricostruzione dell'ambiente reale utilizzato nella condizione sperimentale di confronto diretto con la stanza fisica.

In questa configurazione la ricostruzione viene quindi confrontata con la memoria percettiva dell'ambiente fisico osservato direttamente dai partecipanti, consentendo di analizzare la plausibilità della scena ricostruita rispetto all'esperienza del mondo reale.

4.1.3 Struttura dell'esperimento

L'esperimento si articola in tre fasi principali.

1. **Screening iniziale.** I partecipanti compilano un questionario preliminare finalizzato alla raccolta di informazioni demografiche e alla caratterizzazione del livello di familiarità con tecnologie digitali, videogiochi e sistemi di realtà virtuale.
2. **Esperienza immersiva.** I partecipanti esplorano le due condizioni sperimentali all'interno dell'ambiente VR. Al termine di ciascuna esperienza viene somministrato un questionario direttamente all'interno dell'ambiente immersivo.
3. **Valutazione finale.** Al termine dell'esperimento viene compilato un breve questionario conclusivo volto a raccogliere una valutazione comparativa delle due condizioni sperimentali.

4.2 Flusso di lavoro dello studio sperimentale

La Figura 4.4 illustra il workflow dello studio sperimentale condotto per valutare l'esperienza di fruizione delle ricostruzioni generate tramite Gaussian Splatting in ambiente immersivo. Il processo parte dalla definizione delle ipotesi di ricerca, sulla base delle quali è stato progettato il questionario utilizzato per raccogliere le valutazioni degli utenti. Successivamente è stato predisposto l'ambiente immersivo contenente le ricostruzioni generate con la pipeline selezionata, che è stato utilizzato durante l'esperimento con gli utenti. Le risposte raccolte sono state quindi analizzate attraverso metriche di qualità dell'esperienza (QoE) e tramite analisi statistiche, al fine di verificare le ipotesi formulate e interpretare i risultati dello studio.



Figura 4.4: Workflow dello studio sperimentale: dalla definizione delle ipotesi alla verifica dei risultati ottenuti dall'esperimento in ambiente immersivo.

4.3 Partecipanti

Lo studio ha coinvolto complessivamente **15 partecipanti** volontari.

Il questionario di screening iniziale include domande relative a **fascia d'età**, **genere** e **titolo di studio**, oltre a una sezione di **screening visivo**. Tutti i partecipanti hanno dichiarato una visione normale o corretta a normale mediante occhiali o lenti a contatto e una percezione dei colori normale.

Oltre ai dati demografici, il questionario ha raccolto informazioni relative alla familiarità dei partecipanti con **visori VR**, **videogiochi** e **strumenti informatici**. Questi dati sono stati utilizzati per caratterizzare il profilo tecnologico del campione e per condurre un'analisi esplorativa delle possibili differenze individuali

nell'esperienza immersiva.

La Tabella 4.1 riassume la distribuzione demografica dei partecipanti coinvolti nello studio.

La Figura 4.5 mostra la distribuzione del titolo di studio dei partecipanti, evidenziando una prevalenza di titoli universitari (lauree triennali e magistrali) all'interno del campione considerato.

Tabella 4.1: Distribuzione demografica dei partecipanti coinvolti nello studio.

Caratteristica	Numero partecipanti
<i>Genere</i>	
Maschio	11
Femmina	4
<i>Titolo di studio</i>	
Diploma di Scuola Superiore	2
Laurea Triennale	6
Laurea Magistrale / Master	5
Dottorato di Ricerca	2

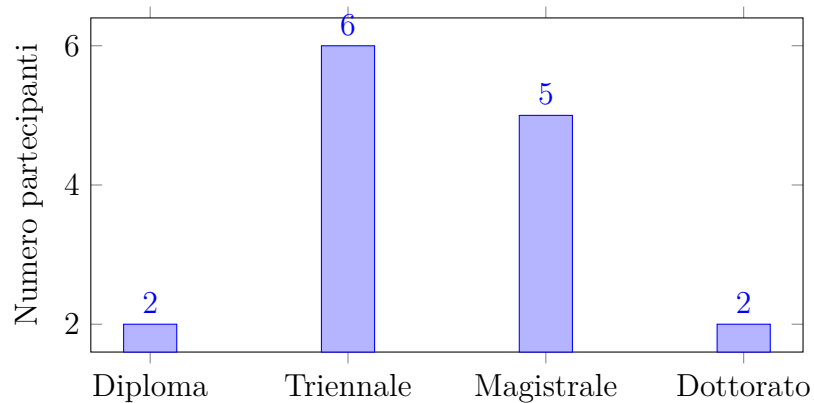


Figura 4.5: Distribuzione del titolo di studio dei partecipanti.

4.4 Ipotesi di ricerca

Sulla base della struttura dell'esperimento e degli obiettivi dello studio sono state formulate alcune ipotesi di ricerca relative alla percezione delle ricostruzioni generate tramite 3D Gaussian Splatting in ambiente immersivo.

Le ipotesi riguardano tre dimensioni principali dell'esperienza utente: la qualità visiva percepita della ricostruzione, il senso di presenza nell'ambiente virtuale e il comfort dell'esperienza immersiva.

- **H1 – Fedeltà visiva della ricostruzione.** La ricostruzione generata tramite 3D Gaussian Splatting presenta un degrado visivo limitato rispetto al riferimento video della scena reale.
- **H2 – Presenza nell’ambiente virtuale.** L’osservazione delle ricostruzioni immersive genera un elevato senso di presenza nello spazio virtuale.
- **H3 – Confronto tra riferimento video e ambiente reale.** Il realismo percepito della ricostruzione differisce tra la condizione di confronto con video e quella di confronto con l’ambiente reale.
- **H4 – Comfort dell’esperienza immersiva.** La visualizzazione delle ricostruzioni non produce livelli elevati di discomfort durante l’esperienza immersiva.
- **H5 – Preferenza degli utenti.** I partecipanti mostrano una preferenza complessiva per una delle due modalità di confronto tra ricostruzione e riferimento (visione del video o osservazione dell’ambiente reale).

Per verificare queste ipotesi sono stati utilizzati questionari standardizzati per la valutazione della qualità visiva, del senso di presenza e del comfort negli ambienti di realtà virtuale.

4.5 Questionari e strumenti di valutazione

Per la raccolta delle valutazioni dei partecipanti sono stati utilizzati questionari derivati da strumenti ampiamente impiegati negli studi di realtà virtuale e di valutazione percettiva.

Le domande sono state progettate con l’obiettivo di misurare quantitativamente le principali dimensioni dell’esperienza utente considerate nello studio: qualità visiva della ricostruzione, senso di presenza nell’ambiente virtuale e comfort dell’esperienza immersiva.

I questionari sono stati somministrati direttamente **all’interno dell’ambiente immersivo** al termine di ciascuna esperienza. Questa modalità consente di raccogliere le valutazioni immediatamente dopo l’osservazione della scena, riducendo possibili interferenze dovute alla memoria o all’interruzione dell’esperienza immersiva.

La Figura 4.6 mostra un esempio della visualizzazione delle domande del questionario all’interno dell’ambiente immersivo utilizzato nell’esperimento.

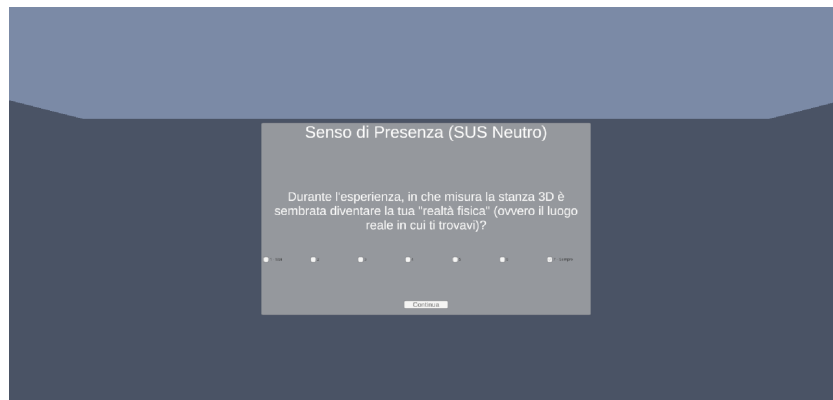


Figura 4.6: Esempio di domanda del questionario visualizzata direttamente all'interno dell'ambiente immersivo.

La qualità visiva percepita della ricostruzione è stata valutata utilizzando il metodo *Double Stimulus Impairment Scale* (DSIS), definito nello standard internazionale per la valutazione soggettiva della qualità delle immagini e dei video [33]. Questo metodo richiede ai partecipanti di valutare il degrado percepito di uno stimolo rispetto a un riferimento osservato immediatamente prima.

Il senso di presenza nell'ambiente virtuale è stato valutato utilizzando alcune domande derivate dal questionario *Slater-Usoh-Steed* (SUS), uno degli strumenti più diffusi per la misurazione della presenza negli ambienti di realtà virtuale [34, 35, 36].

Il comfort dell'esperienza immersiva è stato analizzato tramite domande ispirate al *Cybersickness Questionnaire for Virtual Reality* (CSQ-VR), progettato per valutare il malessere fisico associato all'utilizzo dei visori VR [37]. Questo questionario deriva da strumenti precedentemente utilizzati per la misurazione della simulator sickness negli ambienti virtuali [38, 39].

Le valutazioni di realismo e presenza sono state espresse tramite scale Likert a 7 livelli, dove valori più elevati indicano una valutazione più positiva dell'esperienza.

La qualità visiva percepita è stata invece valutata tramite la scala DSIS a 5 livelli, come previsto dallo standard ITU-R BT.500 [33].

Il livello di cybersickness è stato calcolato come somma delle sei domande derivate dal questionario CSQ-VR. Il valore risultante varia tra 6 e 42, dove valori più elevati indicano un maggiore livello di discomfort durante l'esperienza immersiva.

4.6 Analisi dei dati

Le risposte raccolte tramite i questionari sono state trasformate in metriche quantitative per l'analisi statistica dei risultati. Per ciascuna delle dimensioni considerate

nello studio sono state calcolate statistiche descrittive, in particolare media e deviazione standard delle valutazioni fornite dai partecipanti.

Nel caso della condizione sperimentale basata sul confronto con il video della scena reale (Ambiente A), la qualità visiva della ricostruzione è stata analizzata tramite il *Mean Opinion Score* (MOS) derivato dalla scala DSIS.

Per confrontare le due condizioni sperimentali (confronto con video e confronto con ambiente reale) sono stati utilizzati test statistici non parametrici per dati appaiati. In particolare è stato applicato il test di *Wilcoxon signed-rank*, comunemente impiegato negli studi di valutazione soggettiva in ambito VR. Il livello di significatività statistica adottato per i test è stato fissato a $\alpha = 0.05$.

Per analizzare la possibile relazione tra prestazioni del sistema di rendering e comfort dell'esperienza immersiva è stata inoltre calcolata la correlazione di Spearman tra il frame rate minimo registrato durante l'esperienza e il livello di cybersickness riportato dai partecipanti.

Infine, è stata condotta un'analisi esplorativa delle differenze individuali tra i partecipanti sulla base del livello di familiarità tecnologica dichiarato nel questionario preliminare. I partecipanti sono stati suddivisi in tre gruppi di esperienza (novizi, livello intermedio ed esperti) in relazione alla familiarità con visori VR, videogiochi e strumenti informatici.

La Tabella 4.2 riassume le principali metriche utilizzate per analizzare le diverse dimensioni dell'esperienza utente considerate nello studio.

Tabella 4.2: Metriche utilizzate per l'analisi dei dati dell'esperimento.

Dimensione analizzata	Metrica utilizzata
Qualità visiva percepita	DSIS Mean Opinion Score (MOS)
Realismo percepito	Media delle valutazioni Likert
Senso di presenza	Media delle domande derivate dal questionario SUS
Cybersickness	Somma delle domande CSQ-VR
Affaticamento oculare	Media delle domande oculomotorie CSQ-VR
Prestazioni hardware	FPS minimo e FPS drop
Preferenze degli utenti	Percentuale di scelta nel questionario finale

4.7 Risultati

Questa sezione presenta i risultati ottenuti dall'analisi delle risposte raccolte durante l'esperimento. Le metriche considerate riguardano la qualità visiva percepita delle

ricostruzioni, il senso di presenza nell'ambiente virtuale e il comfort dell'esperienza immersiva.

Le valutazioni di realismo e presenza sono state espresse su scala Likert a 7 livelli, mentre la qualità visiva della ricostruzione è stata valutata tramite la scala DSIS a 5 livelli.

Il livello di cybersickness è stato invece calcolato come somma delle domande derivate dal questionario CSQ-VR.

4.7.1 Quality of Experience

La Tabella 4.3 riassume le principali metriche di Quality of Experience considerate nello studio, in particolare il **realismo** percepito della ricostruzione e il **senso di presenza** nell'ambiente virtuale.

Tabella 4.3: Metriche di Quality of Experience osservate nelle due condizioni sperimentali.

Metrica	Ambiente A (Video vs 3DGS)	Ambiente B (Realità vs 3DGS)
Realismo	5.67 ± 0.90	5.80 ± 1.32
Presenza	5.77 ± 0.94	6.12 ± 0.65
Cybersickness	9.07 ± 4.37	8.33 ± 2.82
Affaticamento oculare	1.80 ± 0.65	1.80 ± 0.80

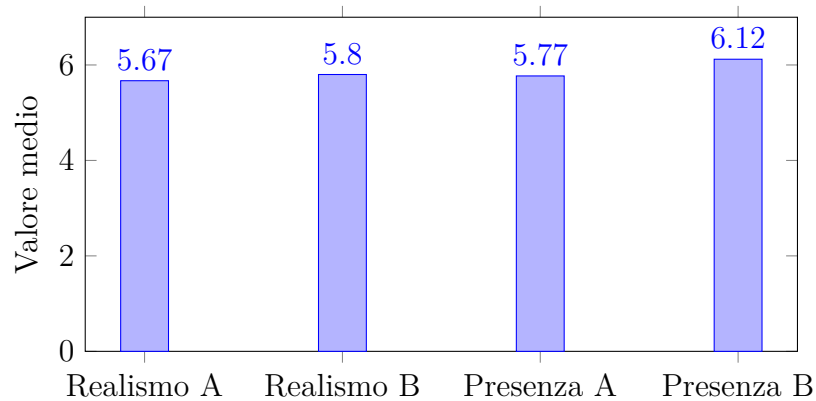


Figura 4.7: Confronto tra realismo percepito e senso di presenza nelle due condizioni sperimentali.

La Figura 4.7 mostra il confronto tra realismo percepito e senso di presenza nelle due condizioni sperimentali. I risultati evidenziano valori medi elevati in entrambe

le configurazioni, con livelli leggermente superiori nella condizione di confronto con l'ambiente reale.

4.7.2 Qualità visiva percepita

Nella condizione sperimentale basata sul confronto tra il video della scena reale e la ricostruzione 3DGS (Ambiente A), la qualità visiva percepita è stata valutata tramite la scala DSIS.

Il valore medio del Mean Opinion Score (MOS) ottenuto risulta pari a

$$MOS = 3.47 \pm 0.92$$

Considerando che la scala DSIS prevede valori compresi tra 1 e 5, il risultato indica che la ricostruzione generata tramite *3D Gaussian Splatting* viene percepita come relativamente vicina al riferimento video, pur presentando un certo livello di degradazione visiva percepita.

4.7.3 Comfort dell'esperienza immersiva

Il comfort dell'esperienza immersiva è stato analizzato tramite le domande derivate dal questionario CSQ-VR.

Nella condizione di confronto con il video (Ambiente A) il livello medio di cybersickness osservato è pari a

$$Sickness_A = 9.07 \pm 4.37$$

mentre nella condizione di confronto con l'ambiente reale (Ambiente B) si osserva un valore medio leggermente inferiore

$$Sickness_B = 8.33 \pm 2.82$$

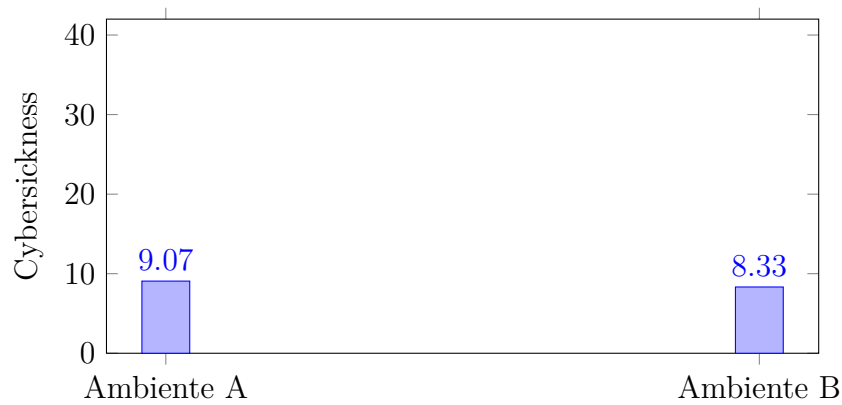


Figura 4.8: Livello medio di cybersickness nelle due condizioni sperimentali.

La Figura 4.8 mostra il livello medio di cybersickness nelle due condizioni sperimentali. I valori osservati risultano relativamente contenuti in entrambe le configurazioni, considerando che la scala CSQ-VR varia tra 6 e 42, indicando un livello complessivamente basso di discomfort durante l'esperienza immersiva.

L'affaticamento oculare medio riportato dai partecipanti risulta pari a

$$Eye\ Fatigue_A = 1.80 \pm 0.65$$

$$Eye\ Fatigue_B = 1.80 \pm 0.80$$

4.7.4 Prestazioni hardware

Durante l'esperienza immersiva sono state registrate anche alcune metriche relative alle prestazioni del sistema di rendering.

Il calo medio di frame rate osservato durante l'esplorazione delle ricostruzioni risulta pari a

$$FPS\ Drop = 75.70 \pm 49.45$$

Per analizzare la possibile relazione tra prestazioni hardware e comfort dell'esperienza immersiva è stata calcolata la correlazione di Spearman tra il frame rate minimo registrato e il livello di cybersickness.

Il coefficiente di correlazione ottenuto è

$$r = 0.428 \quad (p = 0.112)$$

Questo risultato indica una correlazione positiva moderata, che tuttavia non risulta statisticamente significativa nel campione considerato.

4.7.5 Preferenze degli utenti

Al termine dell'esperimento è stata raccolta una valutazione comparativa complessiva tra le due condizioni sperimentali.

La Figura 4.9 mostra la distribuzione delle preferenze dei partecipanti in termini di immersione percepita. La maggioranza dei partecipanti (86.7%) ha indicato come più immersiva la condizione basata sul confronto diretto con l'ambiente reale.

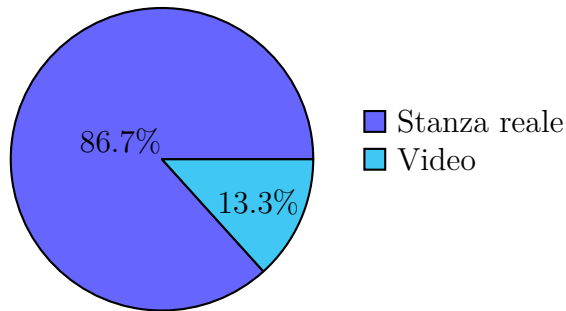


Figura 4.9: Preferenza degli utenti per la condizione percepita come più immersiva.

Per quanto riguarda invece la fedeltà visiva percepita della ricostruzione, il 60% dei partecipanti ha indicato come più convincente la condizione basata sul confronto con la stanza reale.

4.8 Discussione dei risultati

I risultati dell'esperimento consentono di valutare le ipotesi formulate nella sezione precedente relativamente alla percezione delle ricostruzioni generate tramite 3D Gaussian Splatting in ambiente immersivo.

Per quanto riguarda la **fedeltà visiva della ricostruzione**, le valutazioni raccolte indicano che la qualità percepita della ricostruzione risulta complessivamente vicina al riferimento video utilizzato nella prima condizione sperimentale. Questo risultato suggerisce che la tecnica di 3D Gaussian Splatting è in grado di produrre rappresentazioni visive convincenti anche quando osservate in ambiente immersivo.

Dal punto di vista della **presenza percepita**, i partecipanti riportano livelli relativamente elevati di immersione nello spazio virtuale. L'assenza di differenze statisticamente significative tra la condizione basata sul confronto con il video e quella basata sul confronto con l'ambiente reale suggerisce che il senso di presenza dipende principalmente dalla qualità complessiva dell'esperienza immersiva piuttosto che dal tipo di riferimento utilizzato nel confronto.

Anche per quanto riguarda il **realismo percepito**, le valutazioni risultano simili nelle due condizioni sperimentali. Il test statistico di Wilcoxon non evidenzia

infatti differenze significative tra il confronto con il video e quello con l'ambiente reale. Questo indica che la ricostruzione mantiene un livello di plausibilità visiva comparabile sia quando viene confrontata con un riferimento video sia quando viene confrontata con l'ambiente fisico osservato direttamente.

Dal punto di vista del **comfort dell'esperienza immersiva**, i livelli di cybersickness e affaticamento oculare riportati dai partecipanti risultano complessivamente contenuti. Questo risultato suggerisce che la visualizzazione di ricostruzioni basate su 3D Gaussian Splatting non introduce livelli rilevanti di discomfort durante l'esplorazione immersiva.

Infine, l'analisi delle **preferenze degli utenti** evidenzia che la maggior parte dei partecipanti percepisce come più immersiva la condizione di confronto con l'ambiente reale. Questo risultato può essere interpretato come un effetto della memoria percettiva della scena reale osservata immediatamente prima della visualizzazione della ricostruzione.

Nel complesso, i risultati dello studio indicano che le ricostruzioni generate tramite 3D Gaussian Splatting possono offrire un'esperienza immersiva plausibile e relativamente confortevole per l'utente, confermando il potenziale di questa tecnica per applicazioni di rappresentazione immersiva di ambienti reali.

La Tabella 4.4 riassume l'esito delle ipotesi formulate all'inizio dello studio sulla base dei risultati ottenuti.

Ipotesi	Metrica utilizzata	Esito
Fedeltà visiva della ricostruzione rispetto al riferimento video	DSIS MOS	Confermata
Elevato senso di presenza nell'ambiente virtuale	Media questionario SUS	Confermata
Differenze di realismo tra confronto con video e ambiente reale	Realismo percepito + test Wilcoxon	Non confermata
Assenza di discomfort significativo durante l'esperienza immersiva	Cybersickness e affaticamento oculare	Confermata
Preferenza degli utenti tra le due modalità di confronto	Questionario finale	Confermata

Tabella 4.4: Riepilogo della conferma delle ipotesi di ricerca.

L'analisi esplorativa delle differenze individuali suggerisce inoltre che la **familiarità progressiva** con tecnologie digitali, videogiochi e sistemi di realtà virtuale può influenzare alcuni aspetti dell'esperienza immersiva, in particolare la percezione del cybersickness. Tuttavia, a causa della dimensione ridotta del campione, tali risultati devono essere interpretati con cautela e rappresentano principalmente indicazioni preliminari per studi futuri su campioni più ampi.

4.9 Limitazioni dello studio

Nonostante i risultati ottenuti forniscano indicazioni utili sulla percezione delle ricostruzioni generate tramite 3D Gaussian Splatting in ambiente immersivo, lo studio presenta alcune limitazioni che è opportuno considerare nell'interpretazione dei risultati.

Una prima limitazione riguarda la **dimensione del campione**. L'esperimento ha coinvolto un numero relativamente ridotto di partecipanti (15 utenti), tipico degli studi esplorativi in ambito VR ma non sufficiente per trarre conclusioni generalizzabili sull'intera popolazione degli utenti.

Un secondo aspetto riguarda la **tipologia delle scene analizzate**. L'esperimento è stato condotto utilizzando un numero limitato di ambienti indoor. Scene con caratteristiche diverse, come ambienti outdoor o spazi più complessi dal punto di vista geometrico, potrebbero influenzare in modo differente la percezione del realismo e della qualità visiva.

Inoltre, alcune analisi esplorative sono state condotte per valutare l'influenza della familiarità tecnologica dei partecipanti. Tuttavia, la **dimensione ridotta di alcuni gruppi di esperienza** limita la possibilità di trarre conclusioni statisticamente generalizzabili da tali osservazioni.

Nonostante queste limitazioni, lo studio fornisce un contributo sperimentale alla valutazione dell'esperienza utente associata alle ricostruzioni basate su 3D Gaussian Splatting in ambiente immersivo, evidenziando il potenziale di questa tecnica per la rappresentazione immersiva di ambienti reali.

Capitolo 5

Conclusioni e sviluppi futuri

Questo lavoro di tesi ha analizzato il potenziale delle ricostruzioni basate su 3D Gaussian Splatting per la rappresentazione immersiva di ambienti reali all'interno di applicazioni di realtà virtuale. L'obiettivo principale è stato quello di valutare **in che modo integrare questa tecnica all'interno di esperienze di visualizzazione immersiva e in che misura sia in grado di offrire un'esperienza immersiva convincente per l'utente**, considerando sia la qualità visiva percepita delle ricostruzioni sia gli aspetti legati al senso di presenza e al comfort dell'esperienza VR.

Il lavoro ha combinato **un'analisi tecnica degli strumenti di generazione e visualizzazione delle ricostruzioni** con uno **studio sperimentale condotto su utenti reali**, con l'obiettivo di valutare la Quality of Experience associata alla visualizzazione immersiva di scene ricostruite tramite 3D Gaussian Splatting.

5.1 Sintesi del percorso progettuale

Il percorso di ricerca sviluppato in questa tesi si è articolato in due fasi principali.

In una prima fase è stata condotta un'analisi comparativa dei principali strumenti disponibili per il training e la visualizzazione di modelli 3D Gaussian Splatting, con una particolare enfasi sui trainer open source e sui visualizzatori XR. Questa analisi ha permesso di identificare **una pipeline operativa composta dal trainer Inria Gaussian Splatting e dal viewer Unity-VR-Gaussian-Splatting**, selezionati per la stabilità della visualizzazione e per la possibilità di integrazione all'interno del game engine Unity.

In una seconda fase è stato progettato e implementato **un ambiente immersivo dedicato alla sperimentazione con utenti**. All'interno di questo ambiente è stato realizzato uno studio sperimentale volto a valutare la percezione delle ricostruzioni 3D Gaussian Splatting in realtà virtuale.

L'esperimento ha previsto la presentazione di **due ambienti ricostruiti differenti**. Nel primo caso la ricostruzione è stata confrontata con un **video della scena reale** utilizzato come riferimento visivo. Nel secondo caso la ricostruzione è stata invece confrontata con **l'ambiente fisico reale** osservato direttamente dai partecipanti.

5.2 Risultati sperimentali e analisi dei risultati

Lo studio sperimentale ha coinvolto **15 partecipanti** e ha analizzato tre dimensioni principali dell'esperienza utente: la qualità visiva percepita della ricostruzione, il senso di presenza nell'ambiente virtuale e il comfort dell'esperienza immersiva.

I risultati ottenuti indicano che **le ricostruzioni generate tramite 3D Gaussian Splatting sono percepite come visivamente plausibili e in grado di generare un buon livello di presenza nell'ambiente virtuale**. Le valutazioni medie relative al realismo percepito e al senso di presenza risultano relativamente elevate e comparabili tra le due condizioni sperimentali considerate nello studio.

L'analisi statistica dei dati non evidenzia **differenze significative tra la condizione di confronto con il video della scena reale e quella basata sull'osservazione diretta dell'ambiente fisico**. Questo suggerisce che la tecnica di 3D Gaussian Splatting è in grado di mantenere un livello di plausibilità visiva comparabile rispetto a entrambe le modalità di riferimento considerate.

Dal punto di vista del comfort dell'esperienza immersiva, i livelli di cybersickness e di affaticamento oculare riportati dai partecipanti risultano **complessivamente contenuti**. Questo indica che la visualizzazione di scene ricostruite tramite 3D Gaussian Splatting non introduce livelli rilevanti di discomfort durante l'esplorazione immersiva.

5.3 Contributi del lavoro

Il lavoro presentato in questa tesi fornisce diversi contributi nell'ambito della valutazione e dell'utilizzo delle tecniche di 3D Gaussian Splatting per applicazioni di realtà virtuale.

Il contributo principale riguarda **l'analisi comparativa degli strumenti disponibili per il training e la visualizzazione di ricostruzioni basate su 3D Gaussian Splatting**. Negli ultimi anni sono stati proposti numerosi framework e implementazioni della tecnica, spesso sviluppati con obiettivi differenti (ricerca, rendering offline, visualizzazione interattiva o applicazioni XR). Questa varietà di strumenti può rendere complessa la scelta di una pipeline operativa, soprattutto nel contesto della visualizzazione immersiva.

In questo lavoro è stata quindi condotta **una comparazione sistematica dei principali trainer e visualizzatori disponibili**, con particolare attenzione alla loro integrazione in ambienti di realtà virtuale. L'analisi ha permesso di individuare una pipeline operativa stabile composta dal trainer Inria Gaussian Splatting e dal viewer Unity-VR-Gaussian-Splatting, adatta alla visualizzazione immersiva all'interno del motore grafico Unity.

Un ulteriore contributo consiste nella **progettazione di un ambiente sperimentale VR dedicato alla valutazione dell'esperienza utente**. Questo ambiente è stato utilizzato per analizzare la percezione delle ricostruzioni 3DGS in condizioni immersive, considerando diversi riferimenti della scena reale.

Infine, lo studio sperimentale condotto con utenti ha fornito **dati empirici sulla Quality of Experience associata alla visualizzazione immersiva di ricostruzioni 3D Gaussian Splatting**. I risultati ottenuti offrono indicazioni utili per comprendere il potenziale di questa tecnica nella rappresentazione immersiva di ambienti reali.

5.4 Sviluppi futuri

Il lavoro presentato apre diverse possibili direzioni di sviluppo per studi futuri.

Una prima linea di ricerca riguarda l'estensione dello studio sperimentale a **campioni di utenti più ampi e diversificati**. Un numero maggiore di partecipanti permetterebbe di analizzare in modo più approfondito l'influenza di fattori individuali, come l'esperienza pregressa con la realtà virtuale o la familiarità con ambienti digitali.

Un secondo possibile sviluppo riguarda l'analisi di **tipologie di scene differenti**. L'esperimento condotto in questo lavoro si è concentrato principalmente su ambienti indoor; l'utilizzo di scene outdoor o di ambienti caratterizzati da maggiore complessità geometrica potrebbe fornire ulteriori informazioni sulla qualità percepita delle ricostruzioni.

Un ulteriore ambito di sviluppo riguarda l'**ottimizzazione delle prestazioni di rendering nei sistemi VR**. L'integrazione di tecniche di ottimizzazione, come strategie di streaming o metodi di livello di dettaglio dinamico, potrebbe migliorare ulteriormente la stabilità del frame rate e la qualità complessiva dell'esperienza immersiva.

Infine, un possibile sviluppo riguarda l'integrazione delle ricostruzioni 3D Gaussian Splatting all'interno di **applicazioni XR più complesse**, che includano elementi interattivi o ambienti virtuali ibridi in cui scene ricostruite dal mondo reale vengono combinate con contenuti generati proceduralmente.

5.5 Considerazioni finali

Nel complesso, i risultati ottenuti in questa tesi evidenziano **il potenziale della tecnica 3D Gaussian Splatting per la rappresentazione immersiva di ambienti reali** all'interno di applicazioni di realtà virtuale.

Le ricostruzioni generate tramite questa tecnica risultano in grado di offrire **un'esperienza visiva plausibile e un buon livello di presenza nell'ambiente virtuale**, mantenendo al tempo stesso **livelli contenuti di discomfort** durante l'esplorazione immersiva.

Questi risultati suggeriscono che le tecniche di ricostruzione neurale basate su gaussiane possono rappresentare **uno strumento promettente per la creazione di ambienti virtuali fotorealistici** e per lo sviluppo di nuove forme di esperienze immersive basate sulla digitalizzazione di spazi reali.

Bibliografia

- [1] J. D. Foley, A. van Dam, S. K. Feiner e J. F. Hughes. *Computer Graphics: Principles and Practice*. Reading, MA: Addison-Wesley, 1996 (cit. alle pp. 1, 2).
- [2] Z. Wang. «3D Representation Methods: A Survey». In: *arXiv:2410.06475* (2024) (cit. a p. 1).
- [3] J. L. Schönberger e J.-M. Frahm. «Structure-from-Motion Revisited». In: *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*. 2016 (cit. alle pp. 3, 8).
- [4] A. Tewari et al. «Advances in Neural Rendering». In: *arXiv:2304.10050* (2023) (cit. a p. 3).
- [5] B. Mildenhall, P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi e R. Ng. «NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis». In: *Proc. European Conf. Computer Vision (ECCV)*. 2020 (cit. a p. 3).
- [6] B. Kerbl, G. Kopanas, T. Leimkühler e G. Drettakis. «3D Gaussian Splatting for Real-Time Radiance Field Rendering». In: *ACM Transactions on Graphics* (2023) (cit. alle pp. 4, 9–12).
- [7] B. Kerbl, G. Kopanas, T. Leimkühler e G. Drettakis. *3D Gaussian Splatting for Real-Time Radiance Field Rendering*. Project page. 2023. URL: <https://repo-sam.inria.fr/fungraph/3d-gaussian-splatting/> (cit. alle pp. 13, 69).
- [8] MrNeRF. *Awesome 3D Gaussian Splatting*. GitHub repository. 2024. URL: <https://github.com/MrNeRF/awesome-3d-gaussian-splatting> (cit. a p. 13).
- [9] Jawset. *Postshot User Guide*. Official documentation. 2024. URL: <https://www.jawset.com/docs/d/Postshot+User+Guide> (cit. a p. 14).
- [10] Polycam. *Polycam Gaussian Splatting*. Official documentation. 2024. URL: <https://poly.cam/tools/gaussian-splatting> (cit. a p. 14).

-
- [11] Luma AI. *Luma AI Interactive Scenes*. Official documentation. 2024. URL: <https://lumalabs.ai/interactive-scenes/> (cit. a p. 15).
- [12] KIRI Engine. *3DGS Scan Documentation*. Official documentation. 2024. URL: <https://docs.kiriengine.app/category/3dgs-scan> (cit. a p. 15).
- [13] Varjo. *Teleport Documentation*. Official documentation. 2024. URL: <https://teleport.varjo.com/docs/> (cit. a p. 15).
- [14] GraphDeco INRIA. *3D Gaussian Splatting Reference Implementation*. GitHub repository. 2023. URL: <https://github.com/graphdeco-inria/gaussian-splatting> (cit. alle pp. 15, 17).
- [15] Nerfstudio Project. *gsplat: CUDA Accelerated Gaussian Splatting Rasterization*. GitHub repository. 2024. URL: <https://github.com/nerfstudio-project/gsplat> (cit. a p. 16).
- [16] Nerfstudio Project. *Nerfstudio*. GitHub repository. 2024. URL: <https://github.com/nerfstudio-project/nerfstudio> (cit. alle pp. 16, 17).
- [17] Pierotofy. *OpenSplat*. GitHub repository. 2024. URL: <https://github.com/pierotofy/opensplat> (cit. a p. 16).
- [18] MrNeRF. *LichtFeld Studio*. GitHub repository. 2024. URL: <https://github.com/MrNeRF/lichtfeld-studio> (cit. a p. 16).
- [19] wanmeihuali. *Taichi 3D Gaussian Splatting*. GitHub repository. 2024. URL: https://github.com/wanmeihuali/taichi_3d_gaussian_splatting (cit. a p. 16).
- [20] NYU Systems. *Grendel-GS*. GitHub repository. 2024. URL: <https://github.com/nyu-systems/grendel-gs> (cit. a p. 17).
- [21] INRIA GraphDeco. *SIBR: Simple Interactive Bundle-based Rendering Framework*. Official documentation. 2024. URL: <https://sibr.gitlabpages.inria.fr/> (cit. a p. 17).
- [22] Nerfstudio Project. *Nerfstudio Viewer Documentation*. Official documentation. 2024. URL: https://docs.nerf.studio/quickstart/viewer_quickstart.html (cit. a p. 18).
- [23] PlayCanvas. *SuperSplat Viewer*. GitHub repository. 2024. URL: <https://github.com/playcanvas/supersplat-viewer> (cit. a p. 18).
- [24] antimatter15. *splat: WebGL Gaussian Splatting Viewer*. GitHub repository. 2023. URL: <https://github.com/antimatter15/splat> (cit. a p. 18).
- [25] PlayCanvas. *SuperSplat*. GitHub repository. 2024. URL: <https://github.com/playcanvas/supersplat> (cit. a p. 18).
- [26] buaacyw. *GaussianEditor*. GitHub repository. 2024. URL: <https://github.com/buaacyw/GaussianEditor> (cit. a p. 18).

-
- [27] A. Pranckevičius. *UnityGaussianSplatting*. GitHub repository. 2024. URL: <https://github.com/aras-p/UnityGaussianSplatting> (cit. a p. 18).
- [28] M. Kellogg. *GaussianSplats3D*. GitHub repository. 2024. URL: <https://github.com/mkkellogg/GaussianSplats3D> (cit. a p. 18).
- [29] YHK. *Unreal Engine Gaussian Splatting Plugin*. GitHub repository. 2024. URL: https://github.com/YHK-UEPlugins-Public/018_UEGaussianSplatting_Public (cit. a p. 19).
- [30] ninjamode. *Unity VR Gaussian Splatting*. GitHub repository. 2024. URL: <https://github.com/ninjamode/Unity-VR-Gaussian-Splatting> (cit. a p. 19).
- [31] clarte53. *GaussianSplattingVRViewerUnity*. GitHub repository. 2024. URL: <https://github.com/clarte53/GaussianSplattingVRViewerUnity> (cit. a p. 19).
- [32] DL3DV. *DL3DV-10K-Sample Dataset*. HuggingFace dataset repository. 2024. URL: <https://huggingface.co/datasets/DL3DV/DL3DV-10K-Sample> (cit. a p. 26).
- [33] International Telecommunication Union. *Methodology for the Subjective Assessment of the Quality of Television Pictures*. Recommendation ITU-R BT.500-13. ITU, 2012 (cit. a p. 75).
- [34] M. Usoh, E. Catena, S. Arman e M. Slater. «Using Presence Questionnaires in Reality». In: *Presence: Teleoperators and Virtual Environments* 9.5 (2000), pp. 497–503 (cit. a p. 75).
- [35] M. Slater e A. Steed. «A Virtual Presence Counter». In: *Presence: Teleoperators and Virtual Environments* 9.5 (2000), pp. 413–434 (cit. a p. 75).
- [36] M. Slater, A. Steed, J. McCarthy e F. Maringelli. «The Influence of Body Movement on Subjective Presence in Virtual Environments». In: *Human Factors* 40.3 (1998), pp. 469–477 (cit. a p. 75).
- [37] P. Kourtesis, J. Linnell, R. Amir, F. Argelaguet e S. E. MacPherson. «Cybersickness in Virtual Reality Questionnaire (CSQ-VR): A Validation and Comparison against SSQ and VRSQ». In: *Virtual Worlds* 2.1 (2023), pp. 16–35 (cit. a p. 75).
- [38] R. S. Kennedy, N. E. Lane, K. S. Berbaum e M. G. Lilienthal. «Simulator Sickness Questionnaire: An Enhanced Method for Quantifying Simulator Sickness». In: *International Journal of Aviation Psychology* 3.3 (1993), pp. 203–220 (cit. a p. 75).
- [39] A. Singla, S. Göring, D. Keller, R. R. R. Rao, S. Fremerey e A. Raake. «Assessment of the Simulator Sickness Questionnaire for Omnidirectional Videos». In: *Proc. IEEE Virtual Reality (VR)*. 2017 (cit. a p. 75).