UNIVERSITÀ
DEGLI STUDI
DI PADOVA

DIPARTIMENTO
DI INGEGNERIA
DELL'INFORMAZIONE

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

CORSO DI LAUREA IN INGEGNERIA INFORMATICA

"STUDIO DELLA FATTIBILITÁ DEL CONTROLLO DI LATENZA TRAMITE ESTRAPOLAZIONE NELLO STREAMING VIDEO"

Relatore: Prof. / Dott MARCO CAGNAZZO

Laureando/a: MARCO SQUARCINA

ANNO ACCADEMICO 2022 – 2023

Data di laurea 27/09/2023

Alla mia famiglia e
a tutti coloro che mi hanno
sempre supportato!

Abstract

Questo elaborato si concentra sul ruolo cruciale del tempo e della qualità nella trasmissione video, con un'attenzione particolare alla riduzione della latenza Glass-to-Glass (G2G). Questa tesi esplora la possibilità di utilizzare tecniche di estrapolazione video, un'area emergente che offre soluzioni innovative per prevedere futuri frame video. Sebbene l'approccio sia promettente, sono necessari ulteriori sviluppi, in particolare per le previsioni a lungo termine e i tempi di elaborazione. L'analisi si estende all'esame di vari metodi di estrapolazione, ognuno con i propri punti di forza e debolezza, focalizzandosi non solo sull'efficienza nell'elaborazione dei frame, ma anche sull'equilibrio tra tempo e qualità. In definitiva, l'elaborato mira ad analizzare gli aspetti trascurati dalla letteratura scientifica.

Indice

1 Introduzione

Il tempo e la qualità della trasmissione dei dati rappresentano un punto nodale nelle ricerche riguardanti la trasmissione video, un settore in rapida evoluzione che tocca molteplici aspetti della nostra vita quotidiana. In tutte le applicazioni che si occupano di telepresenza, controllo remoto, conferenze virtuali o intrattenimento a distanza, è fondamentale garantire tempi di trasmissione brevi e una Quality of Experience (QoE) soddisfacente.

La necessità di ridurre la latenza identificata dal delay Glass-to-Glass (G2G), ovvero il tempo che intercorre tra l'acquisizione di un frame video e la visualizzazione del medesimo nel terminale remoto ricevente, è una sfida che richiede l'attenzione degli ingegneri e dei ricercatori. Si studia, quindi, la possibilità di utilizzare tecniche di estrapolazione video, un campo emergente che promette soluzioni innovative.

Per estrapolazione si intende il processo per il quale è possibile prevedere futuri frame video, dati i frame già esistenti. Questa tecnica, se ben implementata, potrebbe rappresentare un'alternativa per ridurre la latenza in una trasmissione video, agendo direttamente sul delay G2G. Tuttavia, questo approccio è ancora in fase di sviluppo e richiede miglioramenti significativi per quanto riguarda le previsioni a lungo termine e i tempi di elaborazione, che rimangono delle sfide aperte nel campo.

L'estrapolazione può essere effettuata mediante diversi metodi, ognuno specificamente progettato per la tipologia di video trasmesso. Questo aspetto è di vitale importanza per la qualità dei risultati e verrà trattato di seguito in dettaglio, evidenziando le varie tecniche e i relativi punti di forza e debolezza.

Nei documenti scientifici e nelle ricerche precedenti, si è spesso dato maggior peso nel confronto dei frame, ricavati dalle diverse tecniche, mettendo in secondo piano i tempi di elaborazione. Questo ha creato una lacuna nella comprensione completa dell'efficacia delle varie tecniche di estrapolazione.

In questo elaborato, invece, si andrà ad analizzare, per le tecniche considerate, la fattibilità dell'estrapolazione da diversi punti di vista, mettendo in luce non solo l'efficienza nell'elaborazione dei frame ma anche l'equilibrio tra tempo e qualità. Sarà un'analisi approfondita che mira a fornire una visione chiara del panorama attuale e

futuro dell'estrapolazione video, un campo che promette di rivoluzionare il modo in cui percepiamo e interagiamo con i contenuti video.

1.1 Panoramica dell'elaborato

Nella panoramica dell'elaborato, si intende introdurre ogni capitolo riassumendo il suo contenuto.

Il *primo capitolo* presenta gli obiettivi dell'elaborato e le modalità adottate per raggiungerli.

Il *secondo capitolo* tratta della revisione bibliografica, analizzando gli articoli presi in considerazione e soffermandosi sugli aspetti chiave dell'estrapolazione video. L'obiettivo di questo capitolo è introdurre e descrivere tutti gli elementi necessari per comprendere la panoramica dello studio sulla fattibilità del controllo di latenza tramite estrapolazione nello streaming video.

Il *terzo capitolo* illustra i metodi adottati per analizzare i singoli articoli. Da ogni testo della letteratura scientifica su l'estrapolazione video, sono stati identificati i punti chiave per valutare i risultati ottenuti e gli aspetti da approfondire in futuro.

Il *quarto capitolo* rappresenta la parte cruciale dell'elaborato. In esso vengono descritti e analizzati gli articoli considerati, spiegando lo scopo di ogni ricerca e le modalità di sperimentazione. Questa analisi dettagliata prepara il terreno per le conclusioni che saranno trattate nel quinto capitolo.

Il *quinto* ed ultimo *capitolo* sintetizza le opinioni e i risultati finali dell'elaborato. Dopo aver esaminato i testi relativi ai vari studi sull'estrapolazione video, vengono presentate le conclusioni finali. L'obiettivo di questo capitolo è fornire una visione complessiva sulla tematica trattata nell'elaborato, ovvero lo studio della fattibilità del controllo di latenza tramite estrapolazione nello streaming video.

2 Revisione bibliografica

2.1 Latenza

Con latenza si intende il ritardo di tempo che intercorre tra l'inizio di un evento (come l'invio di un segnale o la richiesta di una risorsa) e l'inizio dell'effetto desiderato o della risposta.
Nello streaming video si intende il ritardo totale che si verifica dal momento in cui un frame viene catturato, elaborato, trasmesso, decodificato e infine visualizzato sul dispositivo dell'utente finale. Esso comprende tutti i ritardi introdotti dai vari componenti del sistema di streaming, come l'elaborazione del segnale, la compressione, la trasmissione attraverso la rete, il buffering e la decodifica.

2.2 Componenti della Latenza

Come si è già detto la latenza nello streaming video ha diverse componenti:

1. **Latenza di Acquisizione:** Il tempo necessario per catturare e inizialmente elaborare il video dalla sorgente.
2. **Latenza di Codifica:** Il tempo impiegato per comprimere il video utilizzando un codec[1] specifico.
3. **Latenza di Rete:** Include i ritardi introdotti dalla trasmissione dei dati attraverso la rete, che può essere influenzata dalla congestione, dalla qualità del collegamento, dalla distanza e altri vari valori.
4. **Latenza di Buffering:** Il tempo di attesa mentre il video viene temporaneamente memorizzato nel buffer prima della riproduzione, per garantire una riproduzione fluida.
5. **Latenza di Decodifica:** Il tempo necessario per codificare il video compresso nel dispositivo utente.

[1] ***Codec:*** Indica il tipo di software utilizzato per la compressione e/o la decompressione di un video digitale.

2.3 Glass-to-Glass delay (G2G)

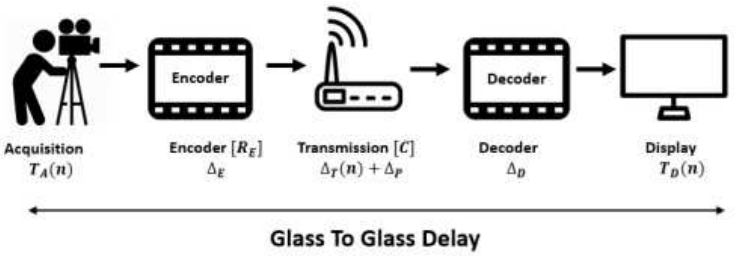


Fig. 1: Latenza G2G in uno schema di trasmissione punto-a-punto [4]

L'insieme delle latenze nello streaming video viene definito come Glass-to-Glass delay [27, 28], ovvero il ritardo tra l'acquisizione di un frame video e la sua visualizzazione al ricevitore.

Il G2G Delay è un fattore critico in molte applicazioni, specialmente quelle che richiedono una comunicazione in tempo reale come le trasmissioni dal vivo, le videoconferenze, e i giochi online. Un eccessivo G2G Delay può portare a una percezione di scarsa sincronizzazione, ritardi nella comunicazione, e una complessiva diminuzione dell'esperienza utente.

Il valore accettabile di G2G Delay nelle conferenze video e nei videogiochi deve essere inferiore a 100 ms per essere sotto la soglia della percezione umana [1], e ritardi ancora minori (10 - 20 ms) per quanto riguarda alle applicazioni che trattano le iterazioni con i macchinari [2]. Tuttavia, il minimo valore del ritardo G2G raggiungibile (attualmente in alcuni scenari applicativi tra 50 e 400 ms [3]) sono limitati inferiormente dai valori di: acquisizione, trasmissione, decodifica e ritardi di buffering[2] [4].

Negli ultimi decenni, sono stati dedicati notevoli sforzi all'ottimizzazione di ciascuna di queste singole fonti di ritardo. Tuttavia, la latenza minima raggiungibile è ancora limitata da vincoli tecnologici e fisici (il più evidente è la velocità della luce), che rappresentano un limite inferiore invalicabile oltre il quale la latenza non può essere ulteriormente ridotta [5].

[2] ***Buffering:*** Il buffering è un processo che coinvolge il pre-caricamento temporaneo di dati in una memoria intermediaria (o buffer) prima che i dati vengano utilizzati.

2.4 Estrapolazione video

Data l'incapacità di ridurre i ritardi fisici nelle trasmissioni video, una possibile alternativa per ridurre la latenza è quella di predire i frame futuri da quelli già disponibili. L'uso della predizione/estrapolazione non è un approccio nuovo. Infatti è utilizzato in molte applicazioni, quali: realtà virtuale, interfacce tattili, videogiochi online, ecc [5].

L'estrapolazione video sfrutta le tecniche di deep-learning[3] per estrarre le caratteristiche dai fotogrammi già acquisiti al fine di prevedere i fotogrammi futuri [7]. Al posto di trasmettere l'ultimo fotogramma acquisito, viene costruito un fotogramma futuro attraverso l'estrapolazione, codificato, trasmesso e decodificato. Se l'orizzonte di estrapolazione è sufficientemente ampio, il fotogramma estrapolato può essere visualizzato dal ricevitore nel momento in cui il fotogramma corrispondente viene acquisito dal trasmettitore. Ciò porta a una significativa riduzione della latenza G2G percepita [5] definita nella sezione precedente.

2.4.1 Funzionamento

L'estrapolazione video può essere effettuata mediante configurazioni: dal lato dell'encoder, da quello del decoder oppure in entrambi i casi. Le diverse modalità sono state analizzate in [9] e hanno mostrato risultati simili per quanto riguarda la degradazione della qualità [4]. Nella sezione 2.4.3 verrà approfondito questo argomento spiegando nel dettaglio le casistiche.

2.4.2 Parametro h

Nei meccanismi di estrapolazione, il parametro h è noto come orizzonte temporale. Se si estrapola il frame x_{n+h} a partire da x_n, $x_{(n-1)}$, ..., $x_{(n-k+1)}$ all'ora l'orizzonte temporale è h ed il numero di frame di contesto è k.
Più alto è il valore di h, più difficile diventa ottenere una previsione affidabile a causa della formazione di artefatti. L' orizzonte temporale rappresenta il bilanciamento tra la compensazione della latenza e la degradazione della qualità. Sebbene il parametro riferito al numero di frame di contesto dipenda dall'estrapolatore utilizzato, la scelta

[3] ***Deep-Learning:*** Il deep learning è una branca dell'apprendimento automatico che utilizza reti neurali profonde per imparare da una gerarchia di concetti. Si basa su strati di elaborazione per creare informazioni astratte e complesse, e trova applicazione in ambiti come riconoscimento vocale e visione computerizzata. Può apprendere rappresentazioni rilevanti dai dati grezzi senza intervento umano.

dell'orizzonte temporale è basata sull'applicazione e non esiste una relazione diretta tra queste due variabili.

2.4.3 Estrapolazione lato decoder ed encoder

L'approccio seguentemente esaminato è stato introdotto per la prima volta in [5], all'interno del contesto del progetto Zero-Latency Linear Coding finanziato dall'Agenzia Nazionale della Ricerca (ANR, Francia) [29]. Con grande interesse, questo approccio è stato trattato da altri partner del progetto. Di seguito verranno riassunte le caratteristiche principali.

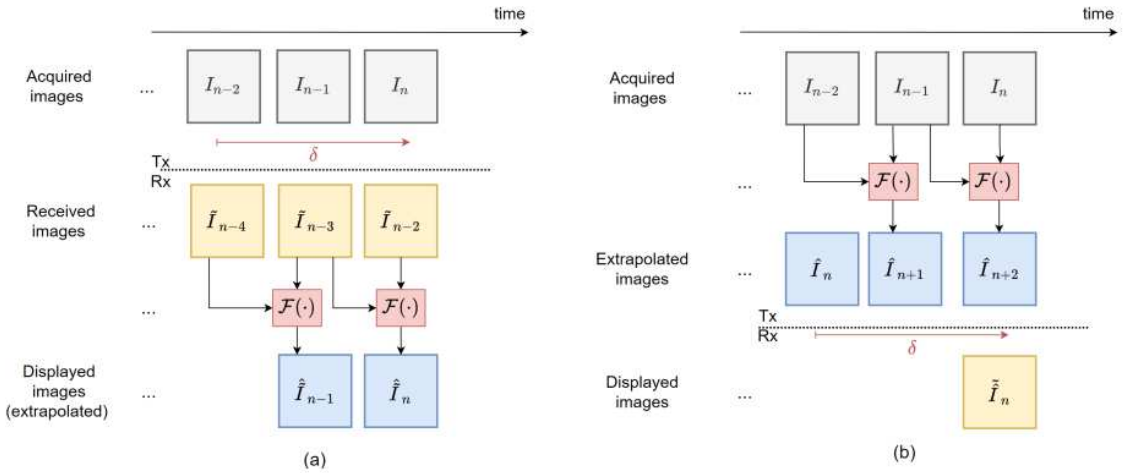


Fig. 2: Un esempio di compensazione tramite (a) estrapolazione dalla parte del decoder e (b) estrapolazione dalla parte dell'encoder. Ĩ e Î indicano rispettivamente i frames decodificati (quantizzati) e predetti (estrapolati). F è la funzione di estrapolazione [5].

In [5] vengono introdotte ed esaminate le tre possibili configurazioni e i loro effetti nei risultati della estrapolazione. Infatti, l'estrapolazione, può essere effettuata: DS (Decoder Side), ES (Encoder Side) o in entrambi i casi. Nelle sezioni successive verranno viste in dettaglio tutte le casistiche.

2.4.3.1 Decoder side

I frame acquisiti sono compressi e trasmessi al ricevitore, dove arrivano con una certa latenza. Come si vede in Fig. 2 (a), I_n rappresenta l' n-esimo frame del video. Ora, per comprendere meglio il funzionamento, si effettua un esempio in cui si assume che la latenza *G2G* δ sia eguale a due frame (corrispondenti al tempo di *2/f* secondi, dove *f* è

pari al frame-rate[4]). Il decoder riceve $\tilde{I}_{n-2}$ (la versione compressa di I_{n-2}) quando l'encoder sta ancora acquisendo il frame I_n.

Per compensare questa latenza, il decodificatore esegue l'algoritmo F per estrapolare i frames che riceve in input dato un numero k di frames decodificati, e produce una previsione $\hat{I}_n$ del frame n come:

$$\hat{\tilde{I}}_n = \mathcal{F}(\{\tilde{I}_{n-h}, \tilde{I}_{n-h-1}, \ldots, \tilde{I}_{n-h-k+1}\}; h) \qquad (1)$$

Per esempio, come si può osservare da Fig. 2 (a), sono considerati K=2 frame di contesto e orizzonte temporale h=2. Ciò significa che si vuole compensare completamente la latenza δ, considerando le assunzioni fatte precedentemente.

Con questo approccio la previsione $\hat{\tilde{I}}_n$ del frame I_n, estrapolata dai frame compressi, può essere visualizzata al decoder mentre il frame I_n viene acquisito dall'encoder.

2.4.3.2 Encoder side

L'estrapolazione ES è illustrata in Fig.2 (b). La descrizione, viene effettuata proseguendo con l'esempio presentato nella sezione precedente.

Il metodo di estrapolazione $\mathcal{F}(-; h)$, è utilizzato con i frames acquisiti: I_n, I_{n-1} ,..., I_{n-k+1} utilizzati come frame di contesto. La previsione dipende dal parametro temporal horizon h, per esempio:

$$\hat{I}(n+h) = \mathcal{F}(\{I_n, I_{n-1}, \ldots, I_{n-k}\}; h) \qquad (2)$$

Come nell'esempio precedente, si ha k=2 e h=2. In seguito i frame estrapolati sono: compressi, trasmessi e visualizzati. In questo caso solo il trasmettitore deve essere adattato e progettato, mentre per ricevere vengono utilizzati dei ricevitori standard.

Nell'esempio attuale, il frame compresso ed estrapolato $\hat{\tilde{I}}_n$ è visualizzato mentre il frame I_n viene acquisito dal trasmettitore [5].

[4] ***Frame rate:*** Il frame rate è la frequenza con cui le immagini consecutive (frame) vengono visualizzate in un video.

2.4.4 Problematiche riguardanti l'estrapolazione

Fig. 3: Sono mostrati due frame corrispondenti allo stesso intervallo temporale. Il frame originale (a) è messo a confronto con il frame (b) estrapolato [4].

L'estrapolazione, oggetto di recenti studi, rappresenta una tecnica molto promettente. La capacità di predire i frame futuri basandosi su quelli già acquisiti potrebbe migliorare la QoE. Tuttavia, ci sono ancora alcuni aspetti da perfezionare.

Come evidenziato in [4], l'estrapolazione incontra difficoltà quando applicata a contenuti video con scene veloci e cambiamenti rapidi di scenario. Infatti, in termini di riduzione della qualità, l'estrapolazione sembra più adatta a contenuti con bassa variazione temporale. Una delle principali sfide del metodo di estrapolazione è rappresentata dai cambiamenti improvvisi di scena o dai tagli: il fotogramma previsto si basa sui fotogrammi precedenti, e tentare di compensare la latenza attraverso l'estrapolazione durante questi tagli risulta impossibile.

Come mostrato in Fig. 3 [4], quando si affrontano scene con rapidi movimenti, l'estrapolazione compromette la qualità dell'immagine in quel punto, a causa della scarsità di informazioni relative a un cambio così repentino.

Un'altra problematica da tenere in considerazione, spesso sottovalutata nell'implementazione di strategie di estrapolazione, è quella riguardante la complessità computazionale. Alcuni metodi di estrapolazione, specialmente quelli che utilizzano tecniche avanzate di analisi del movimento o algoritmi basati su apprendimento profondo, possono essere computazionalmente intensivi e richiedere hardware potente, come GPU dedicate o risorse di calcolo parallelo. Questo non solo limita l'applicabilità su dispositivi meno performanti, come smartphone o computer portatili di fascia bassa, ma può anche aumentare i costi energetici e richiedere una gestione termica ottimale. Inoltre, la necessità di hardware specializzato può creare barriere all'adozione in ambienti con budget limitati o in applicazioni che richiedono soluzioni di elaborazione in tempo reale, rendendo la scelta dell'algoritmo di estrapolazione una decisione critica che deve bilanciare tra precisione, efficienza e praticità.

Data la vasta gamma di tipologie di video, che possono differire per risoluzione, frequenza dei fotogrammi, dinamicità del contenuto e caratteristiche del movimento, le tecniche di estrapolazione sono spesso specifiche e adattate al video in questione. Queste selezionano metodi appropriati in base all'applicazione, come la conversione della frequenza dei fotogrammi, l'ottimizzazione della fluidità o il recupero di dati mancanti. Tuttavia, una scelta non accurata degli algoritmi, senza considerare fattori cruciali come la complessità del movimento, le transizioni di scena o le specificità del contenuto video, può determinare risultati non ottimali. Questi possono manifestarsi come artefatti visivi, imprecisioni nel tracciamento del movimento e perdita di dettagli, compromettendo l'efficacia dell'estrapolazione e, di conseguenza, l'esperienza visiva dell'utente.

2.4.5 Metodi utilizzati

Come detto precedentemente, quando si applica l'estrapolazione si devono utilizzare i metodi più adatti alla tipologia di video in esame. Questo sicuramente è svantaggioso in situazioni dove il video trasmesso varia nelle sue caratteristiche.

I metodi di predizione possono essere *motion-based*, *pixel-based*, e *fusion-based*. I metodi motion-based mirano ad elaborare gli spostamenti dei pixel dell'immagine che corrispondono agli spostamenti degli oggetti nella scena in questione. I metodi pixel-based non utilizzano rappresentazioni del movimento come optical flow o vettori di movimento. I metodi fusion-based invece combinano i metodi motion e pixel-based.

Negli studi esaminati successivamente si può osservare come, diverse tipologie di metodi vengano messe a confronto per confrontare gli esiti.

2.4.6 Tecniche di Deep Learning nell'Estrapolazione Video

Il deep learning applica modelli neurali profondi e complessi per comprendere le relazioni sottili tra i fotogrammi consecutivi in una sequenza video. Le Reti Convoluzionali (CNN) sono spesso utilizzate per catturare le relazioni spaziali tra i pixel, permettendo di analizzare l'intera immagine e prevedere accuratamente i movimenti e le transizioni. In parallelo, le Long Short-Term Memory (LSTM), una specie di reti neurali ricorrenti, hanno dimostrato di essere particolarmente efficaci nel catturare le relazioni temporali nei video, memorizzando informazioni rilevanti da fotogrammi passati per prevedere accuratamente i fotogrammi futuri.

Un altro approccio innovativo è rappresentato dalle Generative Adversarial Networks (GAN), che possono essere utilizzate per generare fotogrammi realistici. Le GAN mettono in competizione due reti, una che genera i fotogrammi e l'altra che li discrimina, per affinare continuamente la qualità dei fotogrammi generati.

Infine, ci sono tecniche basate sulla fusione che combinano vari approcci, come metodi basati su pixel e movimento. Questi metodi integrano diversi livelli di informazione, fornendo una visione più completa e risultati ottimizzati nell'estrapolazione video.
La combinazione di queste tecniche ha aperto nuove strade nel campo dell'estrapolazione video, permettendo una maggiore precisione e la possibilità di affrontare problemi complessi che erano precedentemente insormontabili con metodi tradizionali.

2.4.6.1 Sfide e Limitazioni

Sebbene l'uso del deep learning nell'estrapolazione video sia promettente, presenta alcune problematiche significative. Una di queste è la complessità computazionale, poiché le reti profonde richiedono molte risorse hardware. Questa esigenza di potenza di calcolo può limitare l'utilizzo di tecniche di estrapolazione avanzate su dispositivi meno potenti, restringendo il campo di applicabilità.

Un'altra problematica che emerge è l'overfitting, che è la tendenza del modello a sovra adattarsi ai dati di addestramento. Questo può portare a risultati meno accurati su dati

non visti, riducendo l'efficacia del modello in scenari reali. Inoltre, emergono problemi legati alla formazione di artefatti visivi indesiderati nei fotogrammi estrapolati. Gli errori nel modello di estrapolazione possono tradursi in imperfezioni visive che compromettono la qualità dell'immagine, rendendo l'output meno realistico e piacevole per chi guarda.
Queste sfide mettono in luce la complessità e le sfumature del campo dell'estrapolazione video basata su deep learning. Identificare e adattarsi a tali sfide sarà fondamentale per il futuro sviluppo e l'innovazione in questo settore dinamico e in continua evoluzione.

2.5 Metriche per valutare la qualità dell'immagine

Quando si cerca di paragonare diversi metodi per l'elaborazione di immagini, si ha la necessità di confrontare i risultati. Perché questo sia possibile, si utilizzano delle metriche specifiche. Quelle maggiormente utilizzate sono PSNR e SSIM.

I metodi di valutazione della qualità dell'immagine possono essere suddivisi in metodi oggettivi e soggettivi [14, 15]. I metodi soggettivi si basano sul giudizio umano e operano senza riferimento a criteri matematici [16]. I metodi oggettivi si basano su confronti utilizzando criteri numerici espliciti [17, 18], e sono possibili diversi riferimenti, come la ground truth[5] o la conoscenza dei dati precedenti espressa in parametri[19-21].

Per MSE (Mean Squared Error) si intende il valore relativo all'errore quadratico medio. Esso indica l'errore dei dati osservati da quelli previsti in un determinato modello.

Dato un'immagine di riferimento f e una certa immagine g, entrambe di dimensione M×N, il PSNR tra f e g è definito da:

$$PSNR(f,g) = 10\log_{10}\left(255^2 / MSE(f,g)\right) \quad (3)$$

dove

$$MSE(f,g) = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}\left(f_{ij} - g_{ij}\right)^2 \quad (4)$$

[5] **Ground Truth:** E' definita come informazione nota per essere reale.

Il valore del PSNR tende all'infinito man mano che l'MSE si avvicina a zero; ciò dimostra che un valore PSNR più alto fornisce una qualità dell'immagine superiore. All'altro estremo della scala, un valore piccolo del PSNR implica elevate differenze numeriche tra le immagini. L'SSIM è una metrica di qualità ben nota utilizzata per misurare la somiglianza tra due immagini. È stata sviluppata da Wang et al. [22], ed è considerata correlata alla percezione della qualità del sistema visivo umano (HVS). Invece di utilizzare i metodi tradizionali di sommatoria degli errori, l'SSIM è progettata modellando qualsiasi distorsione dell'immagine come una combinazione di tre fattori che sono la perdita di correlazione, la distorsione della luminanza e la distorsione del contrasto. L' SSIM è definito come:

$$SSIM(f,g) = l(f,g)c(f,g)s(f,g) \tag{5}$$

dove

$$\begin{cases} l(f,g) = \dfrac{2\mu_f\mu_g + C_1}{\mu_f^2 + \mu_g^2 + C_1} \\ c(f,g) = \dfrac{2\sigma_f\sigma_g + C_2}{\sigma_f^2 + \sigma_g^2 + C_2} \\ s(f,g) = \dfrac{\sigma_{fg} + C_3}{\sigma_f\sigma_g + C_3} \end{cases} \tag{6}$$

Il primo termine in (6) è la funzione di confronto della luminanza, che misura la vicinanza della luminanza media delle due immagini (μ_f e μ_g). Questo fattore è massimo e uguale a 1 solo se $\mu_f = \mu_g$. Il secondo termine è la funzione di confronto del contrasto, che misura la vicinanza del contrasto delle due immagini. Qui il contrasto è misurato dalla deviazione standard σ_f e σ_g. Questo termine è massimo e uguale a 1 solo se $\sigma_f = \sigma_g$. Il terzo termine è la funzione di confronto della struttura, che misura il coefficiente di correlazione tra le due immagini f e g. Si noti che σ_{fg} è la covarianza tra f e g. I valori positivi dell'indice SSIM sono compresi nell'intervallo [0,1]. Un valore di 0 significa nessuna correlazione tra le immagini, e 1 significa che f=g. Le costanti positive C_1, C_2 e C_3 vengono utilizzate per evitare un denominatore nullo [23]. Come si può

osservare in [23] i valori l, c ed s sono calcolati blocco per blocco, poiché sono basati sulla ricostruzione dei pixel vicini.

Quality	PSNR	SSIM
95	39.578	0.96866
90	37.469	0.95371
85	36.3	0.94262
80	35.541	0.9339
75	34.956	0.92656
70	34.557	0.92153
65	34.176	0.91499
60	33.855	0.90994
55	33.56	0.90393
50	33.261	0.89813
45	32.97	0.8926
40	32.603	0.88531
35	32.264	0.87868
30	31.802	0.86771
25	31.243	0.85372
20	30.514	0.83233
15	29.542	0.80236
10	27.923	0.74988
5	24.61	0.6283

Fig. 4 [17]

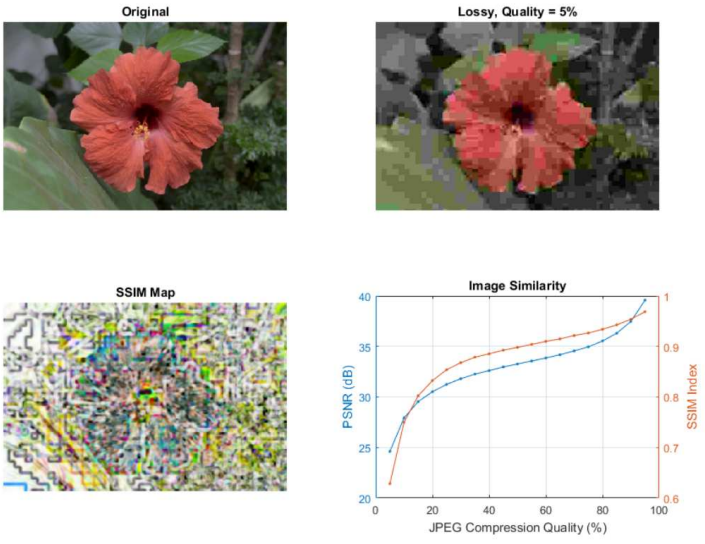


Fig. 5 [17]

Le figure *Fig. 4* e *Fig. 5* indicano uno studio dedicato ad esaminare l'evoluzione dei parametri PSNR e SSIM in base alla compressione applicata ad un'immagine. Maggiore è la degradazione dovuta alla compressione, maggiori sono i valori delle metriche utilizzate.
Nel dettaglio, la Figura 4 contiene tre colonne. La prima colonna, denominata "Quality", mostra in percentuale il valore relativo alla quantità di compressione applicata rispetto al totale. Le altre due colonne, "PSNR" e "SSIM", indicano i valori delle metriche utili per valutare la degradazione delle immagini in output. È osservabile come i valori relativi alla misurazione della qualità seguano l'andamento prestabilito.

Lo studio [25] dimostra come i parametri PSNR e SSIM aumentino, con il diminuire della quantità di compressione applicata e viceversa.

3 Metodologia

Nella letteratura scientifica riguardante l'estrapolazione video, l'attenzione è maggiormente posta sugli esiti dei processi in termini di qualità, trascurando i tempi di elaborazione e la complessità computazionale degli algoritmi. Il rapporto tra il tempo necessario per estrapolare un frame e la qualità del medesimo, influenza quella che poi è la QoE. Quello che questo elaborato mira ad analizzare perciò, è la fattibilità dell'estrapolazione applicata allo streaming video.

Basandosi sugli elementi descritti e analizzati nei capitoli dedicati alla revisione bibliografica, verranno messe a confronto diverse tecniche al fine di fornire una panoramica aggiornata. Questa panoramica si baserà sugli articoli che evidenziano elementi fondamentali per il futuro dell'estrapolazione video. L'obiettivo è analizzare e confrontare i singoli articoli, identificando punti di convergenza o possibili miglioramenti.

3.1 Datasets

I metodi di estrapolazione basati sull'apprendimento necessitano di dati su cui essere addestrati e testati. In questo contesto, i dataset si rivelano fondamentali. Nei casi di studio, come quelli trattati in questo elaborato, i dataset vengono impiegati anche per confrontare le diverse tecniche utilizzate.

Un dataset è una raccolta di dati organizzati e categorizzati. Nel contesto dell'estrapolazione video, i dataset comprendono video di varie tipologie e generi. Più specificamente, sono stati selezionati dataset video con l'obiettivo di testare i metodi in diverse situazioni: si va da video lenti e statici a contenuti con scene più dinamiche e transizioni rapide.

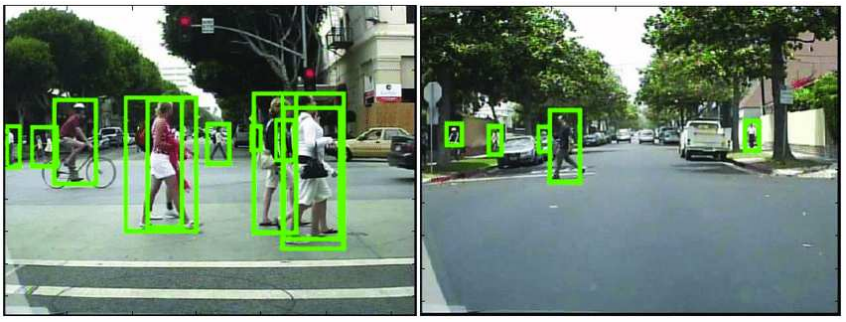


Fig 6: esempio di due frame appartenenti ad un dataset video [26].

In [5], diverse sequenze video sono state considerate per valutare l'influenza delle caratteristiche spaziali e temporali dei video, nei confronti delle tecniche di estrapolazione. Per l'allenamento dei metodi e i test sperimentali, sono stati utilizzati video raffiguranti scenari reali. Il dataset Caltech Pedestrian [30] contiene numerose immagini. Esse provengono da 65 differenti video, registrati a 30fps con una risoluzione di 640x448px. Questo insieme di video, viene utilizzato per effettuare esperimenti su metodi non supervisionati come: MCNet [37] e SDCNet [38].
Per metodi come FlowNet2 [31], che richiedono un pre-allenamento, viene utilizzato il dataset MPI-Sintel [39], il quale contiene ground-truth. L'insieme di video di cui è costituito rappresenta scene veloci, le quali sarebbero critiche per metodi non basati sull'apprendimento.
Per gli esperimenti infine, viene utilizzato un' ulteriore dataset chiamato DriveSeg [32]. Esso è composto da 5000 sequenze catturate a 30fps da un veicolo in movimento, che riprende strade di città affollate.

In [22], oltre ai dataset già trattati, viene introdotto il dataset KTH [33]. È interessante osservare l'uso che viene fatto di queste sequenze video. Infatti, KTH è un dataset contenente video di bassa risoluzione registrati da una videocamera statica con sfondo omogeneo. A differenza dei dataset descritti in precedenza, l'utilizzo di questo tipo di video è determinato dalle caratteristiche del metodo a cui è applicato. Infatti, LMC-Memory [34] richiede molte risorse hardware. L'utilizzo di video a bassa risoluzione permette di ottenere prestazioni migliori.

UFC101 [36] , il quale prende il nome dalle 101 categorie di video da cui è composto, è un dataset creato da video ricavati da YouTube. Mediante l'uso dell'API vengono elaborati i testi dei metadati associati a ciascun video. In questo modo ciascun video viene etichettato e associato ad una categoria.

3.2 Scelte metodologiche

Verranno esaminati e confrontati diversi articoli della letteratura scientifica riguardanti gli argomenti in analisi. Da questi, si cercherà di estrarre i dati necessari per creare una panoramica generale sulla fattibilità del controllo di latenza tramite estrapolazione nello streaming video.

Dal punto di vista tecnico, i parametri confrontati saranno: PSNR e SSIM. Questi parametri, introdotti e descritti nei capitoli precedenti, valutano la qualità dei frame predetti. In aggiunta sarà necessario valutare anche i tempi di elaborazione e i ritardi da essi causati, permettendo in questo modo, di rendere gli studi realistici.

In questo elaborato si cerca di confrontare i metodi esaminati, mediante l'utilizzo degli stessi frames per ogni metodo. Questo permette di osservare gli esiti di tecniche diverse in casistiche uguali. Alcuni articoli, che introducono metodi diversi dagli altri, sono stati testati con altri dataset, questo ovviamente non permette un confronto numerico dei risultati.

4 Sviluppo e analisi

In questo capitolo verranno presentati e analizzati, alcuni articoli della letteratura scientifica riguardanti l'estrapolazione video. L'obiettivo è quello di creare una panoramica tale da trarre delle conclusioni riguardanti la fattibilità del controllo di latenza tramite estrapolazione video.

I principali aspetti su cui occorre concentrarsi sono: la qualità dei risultati, i tempi necessari per l'elaborazione e la complessità computazionale dei metodi utilizzati. Per confrontare i risultati in termini di qualità verranno utilizzate le metriche PSNR e SSIM, già descritte nei capitoli precedenti. Per quanto riguarda i ritardi e la complessità computazionale, sarà necessario trarre conclusioni basate solo sui pochi dati trovati.

4.1 Codifica con Riduzione vs Estrapolazione

Nell'articolo "On the feasability of efficient latency compensation using video frame extrapolation", H. Kanj, A. Trioux, M. Cagnazzo, F.X. Coudoux, P. Corlay, M. Kieffer [4] si esaminano e confrontano i due diversi approcci per la riduzione della latenza video: rate reduction e estrapolazione video. Dati i limiti fisici, per ridurre i ritardi della trasmissione streaming video, si cerca di sviluppare delle strategie a livello computazionale.

4.1.1 Codifica con Riduzione

La codifica con riduzione è un approccio di compressione video elementare, che mira a modificare in modo immediato una componente della latenza. Questo permette di avere video di dimensioni minori riducendo al contempo la larghezza di banda necessaria per la trasmissione.

E' una tecnica semplice, ben nota e padroneggiata in termini di qualità e artefatti generati, offrendo una maggiore flessibilità. Questo è dovuto alla praticità della sua applicazione diretta e alla maggiore granularità, permettendo una precisione più accurata rispetto all' estrapolazione.

Il tasso di codifica è controllato tramite un parametro $\boldsymbol{\alpha} \in]0, 1]$ chiamato fattore di riduzione. R_E [bits/s] indica il tasso di codifica video e $R_E{}'$ [bits/s] il valore ridotto del tasso di codifica video.
Dove:

$$R_E{}' = \boldsymbol{\alpha} R_E \qquad (7)$$

Con Δ_F si indicherà il periodo dei singoli frame in secondi e con C la capacità media del canale di trasmissione.
La latenza risultante in trasmissione è [4]:

$$R_E{}'\Delta_F/C < R_E\Delta_F/C \qquad (8)$$

Dalla formula (8) si osserva come al variare di $\boldsymbol{\alpha}$, si può diminuire la latenza. Questo è possibile intervenendo direttamente riducendo la quantità di dati relativi ai frame codificati .

4.1.2 Estrapolatore scelto

Per l'estrapolazione video viene utilizzata la tecnica SDC-Net [28], date le migliori performance dimostrate in [29]. Il ritardo dell'elaborazione dei frame durante l'estrapolazione dipende dal tipo di architettura utilizzata. Per questo vengono considerate differenti ipotesi del ritardo Δ_x *[s]* dovuto all' estrapolazione: $\Delta_x \in \{0, 1/4, 1/2, 3/4\} \cdot \Delta_F$. Queste ipotesi permettono di esaminare i risultati coprendo un range di possibili diversi tipi di architetture.

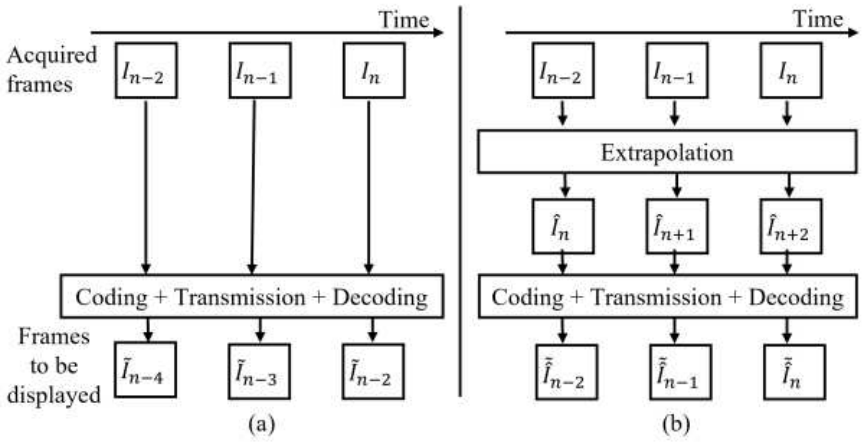


Fig 7: La figura indica tipologie di trasmissione video (a) senza estrapolazione e (b) con estrapolazione. $\hat{I}$ e $\tilde{I}$ indicano rispettivamente i frame estrapolati e decodificati [4].

4.1.3 Analisi del confronto

Nel confronto tra le due tecniche, vengono considerate diverse sequenze video per valutare l'influenza dell'informazione spaziale e temporale [35] sulle prestazioni di entrambi gli approcci di riduzione della latenza. Per informazione spaziale si intende generalmente una misura della quantità di dettagli in un'immagine. Più le scene sono complesse nello spazio, più questo valore è alto. Con informazione temporale invece, si indica la quantità di cambiamenti temporali in una sequenza video. Più questo valore è alto, più movimenti sono presenti in un determinato video. Queste definizioni sono importanti per le sezioni successive. Nei confronti tra le diverse tecniche si tengono in considerazione questi due elementi, poiché la difficoltà relativa alla compressione è direttamente condizionata da entrambi i parametri.

Entrambe le tecniche sono state esaminate al variare dei parametri a loro riferiti. L'estrapolazione dipende dal parametro h temporal horizon, mentre la codifica con riduzione è controllato tramite il fattore di riduzione α.

Dagli esperimenti condotti, emerge che, sebbene vi sia una perdita di qualità dovuta all'estrapolazione, il beneficio ottenuto dalla compensazione della latenza supera quello derivante dalla codifica Rate-Reduction.

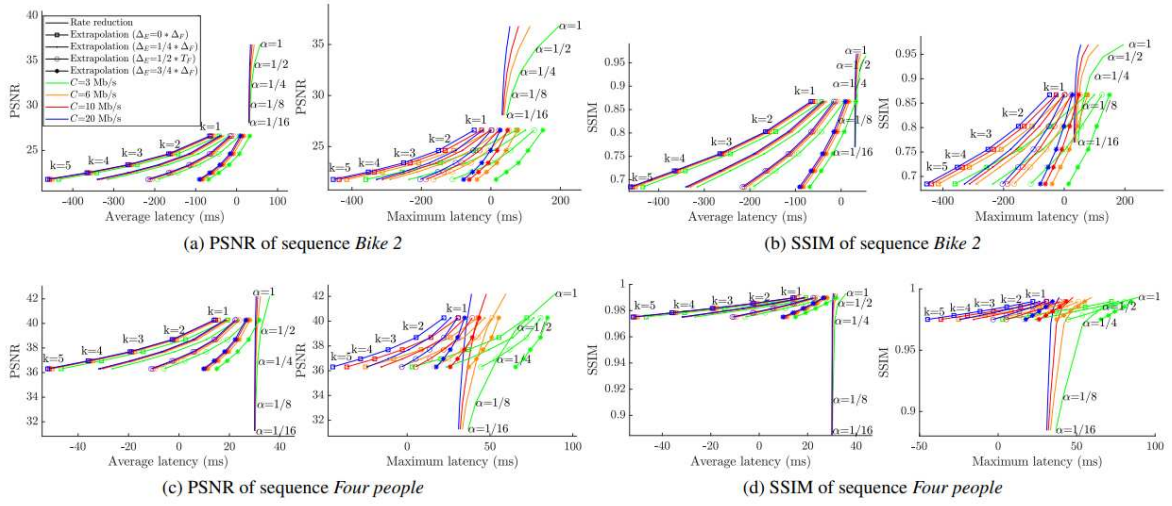


Fig (7,1): PSNR e SSIM in funzione del valore medio e massimo del G2G delay per un $R_E = 800KB/s$ [4].

Nella Fig. 7.1 sono indicati i valori di PSNR e SSIM ottenuti sperimentalmente. Questi dati sono utili a trarre le conclusioni del confronto tra i metodi in esame.

Osservando Fig. 7.1 3c e 3d, si nota che la qualità guadagnata con l'estrapolazione è maggiore. Per esempio, considerando $C = 6Mb/s$ e $\Delta_x = 3/4 \cdot \Delta_F$ per ottenere una latenza massima di 37.3ms, con un orizzonte temporale di estrapolazione pari a h=5, si ottiene un PSNR di 36.5dB e un SSIM di 0.97.
Usando un **α**=⅛ viene ricavato un PSNR di 33.36dB e un SSIM di 0.91 con un guadagno in latenza di 35.6ms.

Confrontando le due tecniche, emerge che l'estrapolazione, sia in termini di tempo di risposta che di qualità per la stessa compensazione di latenza (35 ~ 37ms), supera la codifica con riduzione in termini di qualità. Questo costituisce un'importante indicazione dell'estrapolazione come tecnica promettente. Tuttavia, vi sono ancora numerosi aspetti da perfezionare, tra cui la formazione di artefatti e la complessità computazionale.

4.2 An Online Video Prediction Approach

In questa sezione verrà analizzato l'articolo "All Predictions Matter: an Online Video Prediction Approach", M. Vijayaratnam, M. Cagnazzo, G. Valenzise, E. Tartaglione, del 2023 [21].

Nello streaming video, le tecniche di codifica, decodifica e trasmissione devono essere concepite in modo che dispositivi con caratteristiche diverse possano utilizzarle. A causa delle limitate risorse disponibili, si sviluppano strategie semplici ma funzionali per ridurre la latenza nella trasmissione dei video. Ciò consente di garantire una maggiore QoE per l'utente.
L'articolo [21], propone un metodo che impara in modo continuo (iterativo) dai nuovi dati ricevuti, per migliorare la predizione video. Per questo scopo, viene incorporato uno schema di pesi nella funzione di Loss[6], durante il processo di apprendimento.
La funzione di Loss contiene il parametro λ_i, ovvero il peso della i-esima stima di I_n. $\mathcal{L}^*$ è la somma pesata di tutte le perdite per frame $\mathcal{L}(\hat{I}_n, I_n)$ dove $\hat{I}_n$ è il frame estrapolato e I_n è il frame reale. La funzione è:

$$\mathcal{L}^* = \sum_{i=1}^{h} \lambda_i \mathcal{L}(\hat{I}_n^i; I_n) \quad (9)$$

[6] **Funzione di Loss:** E' una funzione che, tramite il confronto di due frame, indica la perdita di qualità.

4.2.1 Utilizzo della Segmentazione

Il metodo in esame utilizza delle predizioni intermedie tramite i frame ricevuti in input. Ad ogni nuovo frame ricevuto la funzione di estrapolazione $\mathcal{F}$ viene aggiornata.
Il confronto viene effettuato tramite Segmentation Fig. 8. , ovvero una tecnica che focalizza gli oggetti e le loro posizioni nell'intera scena, per poi confrontare queste parti diverse tra i vari frame.

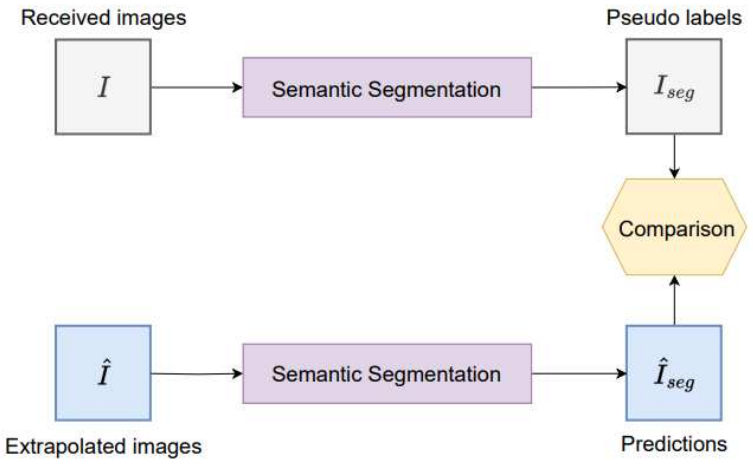
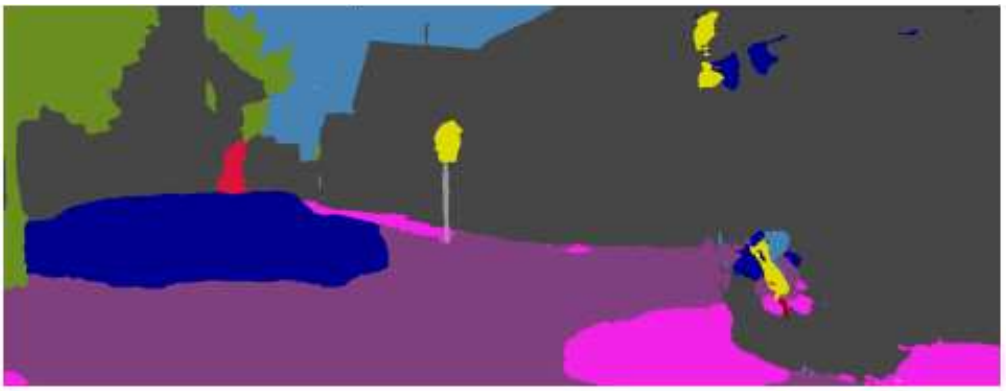


Fig. 8 Metrica basata sulla segmentazione semantica per la previsione video [21].

L'uso della segmentazione mira ad analizzare eventuali errori generati dalla estrapolazione. Per esempio, considerando la segmentazione di un'immagine originale raffigurante una persona o un oggetto, se questi vengono segmentati differentemente dall'immagine estrapolata, si può concludere che la qualità è bassa. Ciò però non è penalizzante per l'estrapolazione, nel caso in cui la segmentazione non riconosca l'eventuale oggetto o persona nell'immagine reale.

(a) Frame estrapolato con SDCNet



(b) Segmentazione dei frame estrapolati

(c) Segmentazione del frame reale

Fig. 9 Output della segmentazione per prevedere un passo nel futuro. Immagine tratta dal dataset Kitti [21].

4.2.2 Conclusioni articolo

Approach	PSNR ↑			SSIM ↑		
	h=1	h=3	h=5	h=1	h=3	h=5
CopyLast	21.25	18.87	17.96	0.50	0.42	0.40
MCNet	23.19	20.66	19.36	0.60	0.52	0.49
FlowNet2 + warp	24.92	21.44	20.03	0.73	0.53	0.48
SDCNet offline	25.38	23.18	22.06	0.76	0.68	0.65
SDCNet online (ours)	**26.53**	**24.07**	**22.73**	**0.83**	**0.75**	**0.71**

(a) Risultati quantitativi utilizzando il dataset Kitti [33] scena 014

Approach	PSNR ↑			SSIM ↑		
	h=1	h=3	h=5	h=1	h=3	h=5
CopyLast	27.65	23.64	22.21	0.72	0.54	0.45
MCNet	28.84	25.20	22.68	0.89	0.74	0.61
FlowNet2 + warp	31.82	27.00	24.72	0.92	0.79	0.65
SDCNet offline	34.23	29.93	28.21	0.95	0.88	0.83
SDCNet online (ours)	**35.89**	**31.71**	**29.66**	**0.98**	**0.93**	**0.89**

(b) Risultati quantitativi utilizzando il dataset DriveSeg [32].

Tabella I: Confronto del metodo proposto online con gli altri metodi di estrapolazione [21].

Utilizzando frame intermedi e conducendo confronti dettagliati tra di loro, ci si imbatte in una complessità di elaborazione notevolmente aumentata. Tuttavia, questo approccio consente di ottenere previsioni di una qualità superiore rispetto ad altri metodi meno sofisticati. Dalla tabella,I (a) e (b), si possono osservare i risultati sperimentali. L'approccio SDCNet [28] online, presentato nell'articolo in esame, dimostra risultati quantitativi maggiori al variare dell'orizzonte temporale rispetto agli altri metodi di estrapolazione.

Data la complessità computazionale, si può dedurre che solo una cerchia ristretta di dispositivi sia in grado di garantire prestazioni soddisfacenti. Tuttavia, è essenziale che i metodi di streaming video siano compatibili con il più ampio numero di dispositivi possibile.
Questo studio evidenzia che, adottando un approccio iterativo e analizzando sequenzialmente i frame intermedi, è possibile affinare le previsioni a lungo termine. La strategia descritta, potrebbe emergere come uno degli approcci più promettenti delle nuove tecniche.

4.3 Un framework di compensazione della latenza per la trasmissione video basato su l'estrapolazione dei frame

In questa sezione verrà analizzato l'articolo "A latency compensation framework for video transmission based on frame extrapolation", M. Vijayaratnam, G. Valenzise, E. Tartaglione, M. Cagnazzo, 2022.

Come già evidenziato in precedenti trattazioni, esistono limiti fisici che determinano quanto si possa effettivamente ridurre la latenza. In questo elaborato, si esplorano in dettaglio le potenzialità e i benefici derivanti dall'uso della tecnica di estrapolazione video. In particolare, l'articolo [22] pone l'accento sull'applicazione pratica dell'estrapolazione nel contesto delle trasmissioni video, cercando di evidenziare le potenziali migliorie e le implicazioni per il settore.
I tentativi precedenti di ridurre la latenza video si sono principalmente concentrati sull' indirizzare singole fonti di ritardo tra quelle menzionate sopra, come la riduzione del tempo di accesso alla rete o del ritardo nella codifica [23], [24]. Tuttavia, questi tentativi presentano due principali svantaggi. In primo luogo, spesso non tengono conto della natura olistica della latenza G2G, che dipende dall'effetto combinato di molteplici ritardi. In secondo luogo, sono intrinsecamente limitati da vincoli fisici, come la velocità del clock[7] dei microprocessori o la velocità della luce, che costituiscono un limite inferiore insuperabile per la latenza minima raggiungibile. Tuttavia, il prezzo da pagare per questo metodo di compensazione della latenza è la potenziale perdita di fedeltà rispetto al video originale a causa dell'estrapolazione video [22]. La possibile degradazione dovuta a l'estrapolazione è raffigurata dalla Fig.3 nella sezione 2.4.4.

4.3.1 Framework proposto per la compensazione della latenza

Il framework proposto considera anche la latenza dovuta al processo di estrapolazione. Questo è un elemento che permette di rendere lo studio più realistico, consentendo di verificare la fattibilità della estrapolazione. In questo caso vengono proposti due diversi schemi, uno con l'estrapolatore dalla parte del codificatore e uno dalla parte del decodificatore.

[7] ***Clock:*** Il clock di un processore è l'elemento che permette di controllare la frequenza dei cicli, con i quali il processore esegue delle istruzioni.

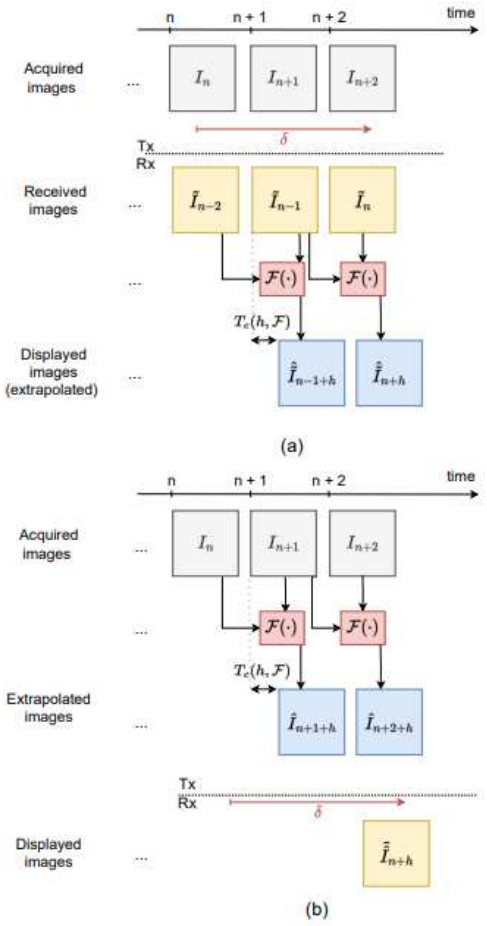


Fig. 10 Schema online per la previsione video in una trasmissione (a) raffigurante il processo di inferenza e (b) il processo di apprendimento [22].

L' obiettivo dell'articolo [22] è quello di compensare la latenza G2G estrapolando i frame futuri (non ancora ricevuti) dai frame disponibili, utilizzando la funzione di estrapolazione Ƒ. Come introdotto nella sezione precedente, dedicata ai parametri relativi all'estrapolazione, vengono considerati i parametri orizzonte temporale h e frame di contesto k.

Il frame predetto è:

$$\hat{I}_{n+h} = \text{Ƒ}(\{I_n, I_{n-1}, \ldots, I_{n-k+1}\}; h) \quad (10)$$

La latenza ridotta **δ** è:

$$\boldsymbol{\delta}(h, T_f, \text{Ƒ}) = h \cdot T_f - T_e(h; \text{Ƒ}) \quad (11)$$

T_f è il periodo di frame, T_e calcola il tempo per estrapolare h frame data la funzione Ƒ.

4.3.2 Adattamento Online per la Trasmissione Video

Nel contesto della trasmissione video, le tecniche tradizionali di estrapolazione vengono generalmente addestrate attraverso una strategia di apprendimento batch, in cui il modello viene addestrato una sola volta. Tuttavia, dato che il vero fotogramma diventa disponibile sul lato del ricevitore dopo un certo lasso di tempo, il modello di estrapolazione può essere aggiornato con l'arrivo di nuovi fotogrammi. Questa metodologia adattiva può portare a previsioni più accurate, stabilendo un equilibrio più favorevole tra latenza e distorsione. Ulteriormente, garantisce una versatilità maggiore adatta alle trasmissioni video. La tecnica di apprendimento online consente di aggiornare in tempo reale un modello generico di estrapolazione video. Ciò assicura una costante adattabilità man mano che nuovi fotogrammi vengono ricevuti durante la trasmissione video [22].

4.3.2.1 Estrapolazione con temporal horizon h = 1

Nell'approccio con h=1 la funzione di estrapolazione $\mathcal{F}$ è considerata fissa durante tutta la sequenza di fotogrammi video. Tuttavia, $\mathcal{F}$ essendo un estrapolatore basato sull'apprendimento, per mantenerlo fisso è necessario ad allenarlo offline con un certo set di dati proveniente dai molteplici dataset disponibili. L'estrapolatore può essere utilizzato appena i k frames, necessari per la previsione, sono disponibili.

La prima estrapolazione è eseguita al tempo $t_k = (k-1)T_f$.
Al tempo $t_k + T_e$ l'estrapolatore produce I_{k+1}. Questo frame è immediatamente inviato al ricevitore. In seguito al tempo $t_{k+1} = t_k + T_f$, la "vera e propria" immagine $\tilde{I}_{k+1}$ è disponibile.
Nell'istante in cui l'immagine reale arriva, è possibile avviare due processi paralleli. Uno che confronta l'immagine estrapolata con l'immagine reale, per permettere all'estrapolatore di apportare delle modifiche utili al miglioramento delle previsioni. Si utilizza la funzione di Loss vista precedentemente. Il secondo continua ad estrapolare i frame richiesti [22].

4.3.2.2 Estrapolazione con temporal horizon h > 1

Il caso in cui il temporal horizon è maggiore di uno, presenta alcuni problemi pratici. Un primo problema riguarda il tempo di inferenza $T_e(h, \mathcal{F})$.La maggior parte delle reti di

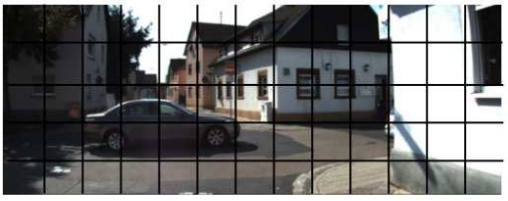
estrapolazione attuali si basa su un metodo iterativo per prevedere l'immagine $I_n + h$. Questo processo richiede un tempo $h \cdot T_e(1, \mathcal{F})$ che cresce proporzionalmente con h, limitando così la quantità di riduzione della latenza che si può ottenere.
Per ovviare a questo problema, si usa per il network di estrapolazione una sequenza sottocampionata:

$$\hat{I}_{n+h} = \mathcal{F}(I_n, I_{n-h}, \ldots, I_{n-(k-1)h}) \quad (12)$$

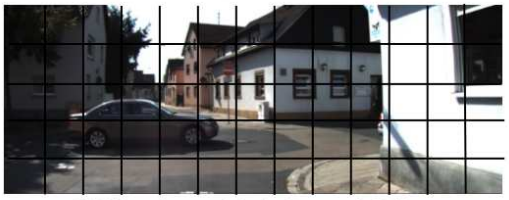
Questo è equivalente ad usare un network di estrapolazione di h=1 (vista in 4.3.2.1) su una sequenza video con un frame-rate $f^* = f/h$ [22].

4.3.3 Conclusioni articolo

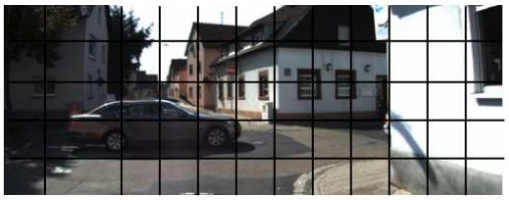
Negli esperimenti condotti, le tecniche in esame sono state analizzate utilizzando diverse metodologie di estrapolazione e vari datasets. Questo al fine di valutare i risultati in differenti contesti. Per l'addestramento degli algoritmi, è stato impiegato il dataset Caltech Pedestrian [30]. Tuttavia, per la valutazione dei risultati, sono stati utilizzati i datasets DriveSeg [32], UCF101 [36] e KTH [33] [22].



(a) Frame originale al tempo t



(b) Ultimo frame originale al tempo (t-100) ms



(c) Estrapolazione (con latenza compensata) con SDCNet

Fig. 10 Risultati qualitativi della compensazione di latenza di 100 ms. E' stato usato lo schema con l'estrapolatore dalla parte del decoder. Per la quantizzazione è stato utilizzato il codec HEVC con parametro di quantizzazione QP=27 [22].

L'articolo ha esplorato la compensazione della latenza nella trasmissione video e l'estrapolazione video. Gli autori hanno presentato un modello semplice di latenza G2G e hanno descritto lo schema proposto di compensazione della latenza. Hanno introdotto una tecnica di adattamento online per sfruttare i fotogrammi decodificati effettivi man mano che vengono ricevuti dal ricevitore. L'articolo conclude sottolineando l'importanza dell'estrapolazione video come bottleneck della soluzione proposta e ha suggerito che ulteriori ricerche dovrebbero concentrarsi direttamente sullo strumento di estrapolazione video stesso.

4.4 Estrapolazione Video nel Tempo e nello Spazio

In questa sezione verrà analizzato l'articolo "Video Extrapolation in Space and Time", Y. Zhang, J. Wu, 2022. La scelta di esaminare questo studio, sebbene non sia confrontabile numericamente con le sezioni precedenti, è mirata ad evidenziare la possibilità di implementare contemporaneamente due tecniche di estrapolazione.

L'articolo [25] analizza la possibilità di utilizzare: Novel View Synthesis (NVS) e video prediction (VP), solitamente distinte, per migliorare l'estrapolazione. Tuttavia, entrambi possono essere visti come modi per osservare il mondo spazio-temporale. L'articolo introduce il concetto di Video Extrapolation in Space and Time (VEST). Il modello proposto utilizza le caratteristiche legate ad entrambe le tecniche VP e NVS, per migliorare l'estrapolazione. Gli esperimenti dimostrano che il metodo proposto supera o è paragonabile a diversi metodi NVS e VP mediante il confronto tramite dataset video.

4.4.1 Novel View Synthesis (NVS)

La Novel View Synthesis (NVS) è una tecnica nella visione artificiale che mira a generare una rappresentazione di una scena da un nuovo punto di vista che non era presente nei dati di input originali. In pratica, NVS cerca di creare una nuova immagine o una sequenza di immagini di una scena come se fosse stata catturata da una posizione o un angolo di telecamera diverso da quelli delle immagini originali. Questo compito è particolarmente impegnativo poiché richiede una comprensione profonda della struttura 3D dell'ambiente e delle relazioni spaziali tra gli oggetti nella scena. La capacità di sintetizzare nuove visualizzazioni ha applicazioni in molte aree, come la realtà virtuale, la realtà aumentata, la cinematografia e la grafica 3D, dove può essere utile o necessario

visualizzare una scena da diverse angolazioni senza la necessità di catturare effettivamente ogni vista possibile con una telecamera [25].

4.4.2 Modello VEST (Video Extrapolation in Space and Time)

Il modello VEST proposto si basa su una rappresentazione generalizzata delle Multiplane Images (MIPs) per eseguire l'estrapolazione spazio-temporale. Questa rappresentazione viene utilizzata per prevedere piani MPI da input monoculare e sfruttare frame storici per l'inferenza del movimento. Il modello può produrre risultati realistici in entrambi gli spazi e l'estrapolazione temporale su una vasta gamma di set di dati.

4.4.2.1 Multiplane Images (MPIs)

Le Multiplane Images (MPIs) sono una rappresentazione stratificata che scompone le immagini in piani RGBA. Originariamente, gli MPIs decompongono le immagini in questi piani, con ogni piano corrispondente a una profondità fissa. Questi piani sono paralleli frontalmente alla telecamera e possono essere descritti da una normale specifica del piano. La rappresentazione MPI è utilizzata per rappresentare un'immagine in termini di questi piani stratificati, ciascuno dei quali ha valori RGB e un valore alfa associato [25].
Inoltre, nel contesto dell'articolo, gli MPIs sono stati generalizzati per modellare anche la dinamica temporale, parametrizzando il campo di flusso di ogni piano. Questo permette di renderizzare immagini da un timestamp futuro e da un nuovo punto di vista attraverso la deformazione basata sul flusso e sull'omografia, rispettivamente [25].

In sintesi, gli MPIs forniscono una rappresentazione stratificata delle immagini, permettendo una comprensione dettagliata della scena in termini di profondità e, quando generalizzati, anche della dinamica temporale.

4.4.2.2 Funzionamento VEST

Il modello riceve come input due fotogrammi consecutivi da una sequenza video denotati come I_{t-1} e I_t. In output viene prodotto una rappresentazione generalizzata delle Multiplane Images (MPIs) per il fotogramma I_t.

Durante l'inferenza, con gli input I_{t-1} e I_t, il modello può essere interrogato per estrapolare ad altre coordinate spazio-temporali. I fotogrammi futuri, con un orizzonte più lungo, come I_{t+2}, I_{t+3}, ecc., possono essere dedotti il modello in modo autoregressivo. Anche quando è disponibile solo un fotogramma di input I_t, il modello può sintetizzare la nuova vista I'_t replicando due volte I_t come input, producendo comunque risultati realistici di NVS. A differenza di altri metodi che si concentrano esclusivamente sull' estrapolazione spaziale, VEST non solo incapsula la sintesi di nuove visualizzazioni (NVS) ma ha anche la capacità di eseguire l'estrapolazione temporale.
L'obiettivo principale di VEST è combinare le informazioni spaziali e temporali per fornire una rappresentazione più completa e dettagliata della scena, permettendo sia l'estrapolazione spaziale che quella temporale [25].

4.4.3 Conclusioni articolo

Nell'articolo [25] si considerano NVS e VP come estrapolazioni lungo due assi per le coordinate spazio-temporali dei video. NVS utilizza i cambiamenti del punto di vista della telecamera in una sequenza video per scoprire la profondità, mentre VP considera sia i movimenti della telecamera che degli oggetti. I due compiti possono essere appresi congiuntamente per sviluppare una rappresentazione della scena dai dati video, con segnali di apprendimento complementari provenienti da ciascuno dei compiti. Proponiamo una rappresentazione MPI generalizzata per affrontare entrambi i compiti e sviluppiamo un modello che raggiunge prestazioni superiori o paragonabili rispetto ai metodi precedenti che affrontano solo uno dei compiti, su set di dati naturali per scene interne ed esterne.

Sebbene, questa tecnica offra una modalità di confronto che migliora l'estrapolazione, sono necessari ulteriori test specifici per confermare l'aumento della complessità computazionale, dovuto all'utilizzo di più approcci combinati. In un ambiente streaming, come quello in esame in questo elaborato, è necessario che le tecniche siano utilizzabili da più dispositivi possibili. Questa tematica necessita ulteriori studi ed esperimenti, per approfondire la tematica riguardante la trasmissione video.

4.5 Analisi risultati

Gli articoli precedentemente analizzati sono stati scelti poiché si soffermano su aspetti che migliorano l'estrapolazione video. Analizzando lo studio di fattibilità del controllo di latenza tramite estrapolazione nello streaming video, l'elaborato mira a raccogliere gli elementi principali di ogni articolo e a constatare similitudini e fattori che migliorano i metodi di estrapolazione.

Dagli studi analizzati, si osserva che un approccio iterativo nell'estrapolazione video migliora la qualità delle previsioni. Come visto in "An Online Video Prediction Approach" [cap 4], confrontare i frame in arrivo con i frame estrapolati permette di aggiornare la funzione F di estrapolazione. Questo assicura un continuo miglioramento man mano che il video viene trasmesso. È importante, tuttavia, tenere conto della complessità computazionale di questo metodo. Quando si parla di streaming video, bisogna considerare macchine con prestazioni limitate. Più l'intervallo iterativo in cui si eseguono i confronti è piccolo, minore è la quantità di dati da gestire per ogni estrapolazione.

Le previsioni effettuate tramite diverse tecniche di estrapolazione hanno dimostrato risultati migliori quando i valori del parametro *temporal-horizon* erano piccoli. Infatti, più i frame da prevedere sono distanti dai frame in esame, maggiore è la probabilità che vengano creati artefatti. Questa situazione si verifica soprattutto quando i video trasmessi contengono scene con elementi in rapido movimento o cambi rapidi di scenografia. In questi video, la quantità di informazione tra un frame e il seguente è bassa rispetto a un video lento.

L'approccio presentato nell'articolo "Video Extrapolation in Space and Time" offre interessanti spunti di riflessione. L'unione di due metodi precedentemente visti garantisce frame estrapolati di qualità superiore. In applicazioni come la guida autonoma, la precisione delle previsioni è un aspetto di fondamentale importanza. Tuttavia, a differenza della trasmissione streaming, queste applicazioni possono contare su elaboratori specifici. Di conseguenza, la complessità che la tecnica VEST comporta non causa problemi nello stesso modo in cui li causerebbe nello streaming video. L'idea di unire diversi metodi per migliorare i risultati rappresenta sicuramente un ottimo spunto per studi futuri.

5 Conclusioni

In questo capitolo, presenteremo le conclusioni tratte dall'intero elaborato. Attraverso l'analisi degli articoli esaminati, ci proponiamo di fornire una panoramica completa riguardo allo studio della fattibilità nella riduzione della latenza nello streaming video.

5.1 Difficoltà incontrate

Una delle principali sfide affrontate durante questo studio è stata la difficoltà nel reperire dati numerici relativi ai vari valori di latenza associati all'uso di tecniche di estrapolazione. La letteratura scientifica, nella sua maggioranza, quando tratta l'argomento dell'estrapolazione, tende a focalizzarsi prevalentemente sulla qualità dei frame estrapolati, trascurando spesso aspetti come i tempi di estrapolazione e la complessità computazionale delle tecniche impiegate. Tuttavia, uno degli articoli che abbiamo esaminato ha integrato valori numerici nei suoi calcoli, cercando di rendere lo studio il più realistico possibile. Questa lacuna nella letteratura può essere in parte giustificata dal fatto che l'estrapolazione è una tecnica relativamente nuova e ancora in fase di perfezionamento. Con l'evoluzione continua della ricerca sulle reti neurali, è ragionevole aspettarsi che emergano nuove tecniche di deep learning specificamente dedicate alla estrapolazione video.

Un altro ostacolo affrontato riguarda il confronto tra le diverse tecniche di estrapolazione. Ogni articolo tende a stabilire il proprio "ambiente" di sperimentazione per effettuare i confronti. Questa pratica, tuttavia, rende complesso il confronto di nuove tecniche con quelle analizzate in studi precedenti. Ogni qualvolta si intende effettuare un nuovo confronto, diventa indispensabile configurare un ambiente sperimentale ad hoc. Questa situazione sottolinea l'importanza e l'urgenza di una standardizzazione degli esperimenti nel campo.

5.2 Suggerimenti per le ricerche future

Dall'analisi di vari articoli dedicati all’estrapolazione video, è evidente una certa carenza nell'attenzione rivolta alla creazione di scenari realistici. È fondamentale che ogni test non si limiti solo a valutare la qualità dei frame estrapolati, ma includa anche altri parametri rilevanti. Considerando l'estrapolazione come una tecnica promettente per

ridurre la latenza nello streaming video, diventa essenziale poter confrontare le tecniche tenendo conto dell'interazione tra qualità, ritardi e complessità computazionale.

5.3 Conclusioni finali

In un contesto futuro in cui la trasmissione video tra dispositivi diventerà sempre più diffusa in ambiti come videoconferenze, controllo remoto di dispositivi, videosorveglianza e piattaforme di condivisione video, è ragionevole prevedere un crescente interesse verso lo studio delle tecniche di estrapolazione video. Questo al fine di garantire a un numero sempre maggiore di utenti la possibilità di beneficiare dei servizi di streaming video, assicurando al contempo un'elevata Qualità dell'Esperienza (QoE).

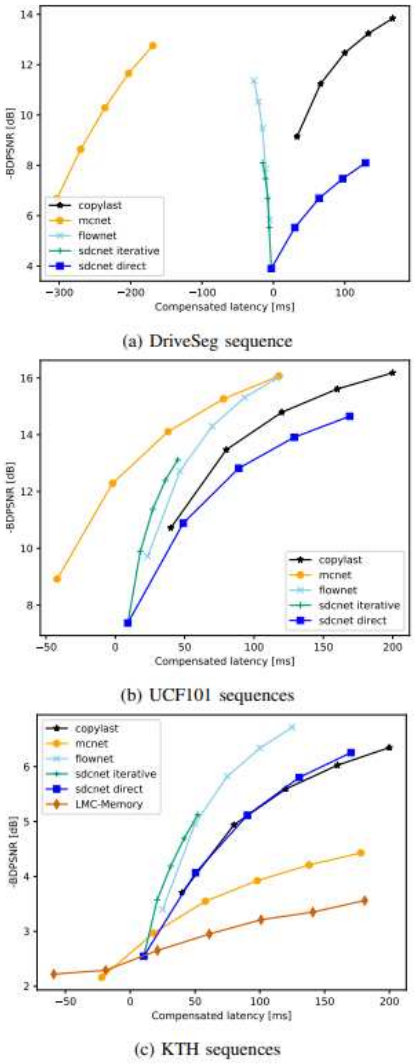


Fig. 11 Compromesso latenza-distorsione con tempo di estrapolazione preso in considerazione utilizzando HEVC LDP [22].

Gli articoli introdotti ed esaminati nelle sezioni precedenti, sono stati scelti per le loro differenze. Ognuno di essi studia approcci diversi, confrontandone i risultati. Emerge l'importanza della corretta relazione tra dataset di allenamento/test e metodo di estrapolazione associato.
Per poter confrontare con una maggiore chiarezza i risultati quantitativi, è necessario descrivere i metodi presi in esame nelle sezioni precedenti. Esponendo le loro peculiarità, sarà possibile comprendere meglio i risultati ottenuti e trarre delle conclusioni adeguate.

Copylast è un metodo che copia semplicemente l'ultimo frame disponibile all' encoder/decoder e lo visualizza in uscita. Proprio per questo non è una vera e propria tecnica di estrapolazione. Esso è utilizzato negli studi come reference ovvero, se un metodo di estrapolazione fornisce risultati inferiori rispetto a Copylast, allora i frame estrapolati contengono un numero elevato di artefatti.
MCNet [37], già nominato nelle sezioni precedenti, è una tecnica pixel-based. Essa permette di ricavare indipendentemente l'informazione spaziale e temporale di un frame

video. Questo è possibile mediante l'uso della memoria a lungo e breve termine, che conserva le differenze tra i frame elaborati [21]. SDCNet [28] è basato sulla fusione tra optical-flow, per derivare l'andamento dei pixel in frame consecutivi, e il kernel convoluzionale. Sia MCNet che SDCNet sono tecniche basate sul pre-allenamento. È importante che i dataset utilizzati per questo scopo, contengano video con elevate informazioni temporali. Questo permette un migliore adattamento a tipologie di sequenze video diverse. In questo elaborato vengono considerati CaltechPedestrian [30] e UFC101 [36].
FlowNet [31] utilizza l'approccio optical-flow per la predizione dei frame futuri. Come ben descritto nella sezione 3.1, necessita l'utilizzo del dataset MPI-Sintel [39]. Alcuni test utilizzano un ulteriore tecnica di estrapolazione chiamata LMC-Memory [40]. Essa estrapola i frame futuri considerando i movimenti dei punti chiave memorizzati a lungo termine.

La Fig.11 indica la rappresentazione dei risultati quantitativi tramite la relazione tra la qualità di un frame estrapolato e la latenza compensata dal metodo in esame. In questo test, viene considerato il tempo T_e relativo alla estrapolazione, usando una scheda video NVIDIA Geforce RTX 3090. I metodi in analisi sono stati testati su tre differenti dataset: DriveSeg[32], UCF101[36] e KTH[33].

	Extrapolation horizon h	-BDPSNR ↓ 1	-BDPSNR ↓ 3	-BDPSNR ↓ 5	-BDSSIM ↓ 1	-BDSSIM ↓ 3	-BDSSIM ↓ 5
HEVC LDP	CopyLast	9.14 (-0.06)	12.47 (-0.02)	13.84 (+0.00)	0.18 (+0.00)	0.34 (+0.00)	0.42 (+0.00)
	MCNet	6.69 (+0.33)	10.29 (+0.29)	12.76 (+0.28)	0.06 (+0.01)	0.16 (+0.00)	0.26 (+0.00)
	FlowNet2+ warping	5.84 (-0.07)	9.47 (-0.06)	11.37 (-0.07)	0.04 (+0.01)	0.14 (+0.00)	0.23 (+0.01)
	SDCNet iter	**3.90 (+0.11)**	**6.69 (+0.10)**	**8.10 (+0.07)**	**0.03 (+0.00)**	**0.08 (+0.00)**	**0.11 (+0.00)**
	SDCNet direct	**3.90 (+0.11)**	7.01 (+0.11)	8.84 (+0.06)	**0.03 (+0.00)**	**0.08 (+0.00)**	0.13 (+0.00)

(a) Risultati quantitativi su DriveSeg

HEVC LDP	CopyLast	10.72 (+0.02)	14.79 (+0.00)	16.18 (+0.00)	0.18 (+0.00)	0.33 (+0.00)	0.38 (+0.00)
	MCNet	8.92 (+0.00)	12.29 (-0.03)	16.07 (-0.04)	0.12 (+0.00)	0.29 (+0.00)	0.35 (+0.00)
	FlowNet2+ warping	9.72 (+0.01)	14.30 (-0.05)	15.99 (-0.05)	0.12 (+0.00)	0.28 (+0.00)	0.36 (+0.00)
	SDCNet iter	**7.37 (+0.16)**	**11.38 (+0.14)**	**13.12 (+0.11)**	**0.08 (+0.01)**	**0.19 (+0.01)**	**0.25 (+0.00)**
	SDCNet direct	**7.37 (+0.16)**	12.82 (+0.12)	14.65 (+0.09)	**0.08 (+0.01)**	0.24 (+0.01)	0.31 (+0.01)

(b) Risultati quantitativi su UFC101

HEVC LDP	CopyLast	3.70 (-0.02)	5.59 (-0.03)	6.35 (-0.04)	0.04 (+0.00)	0.07 (+0.00)	0.09 (+0.00)
	MCNet	**2.16 (-0.10)**	3.55 (-0.13)	4.21 (-0.19)	0.02 (+0.00)	0.04 (+0.00)	0.06 (+0.00)
	FlowNet2+ warping	3.39 (-0.32)	5.82 (-0.41)	6.72 (-0.39)	0.02 (+0.00)	0.06 (+0.00)	0.08 (+0.00)
	SDCNet iter	2.55 (+0.11)	4.19 (+0.19)	5.12 (+0.31)	**0.01 (+0.00)**	0.04 (+0.01)	0.05 (+0.01)
	SDCNet direct	2.55 (+0.11)	5.11 (+0.11)	6.25 (+0.03)	**0.01 (+0.00)**	0.05 (+0.00)	0.07 (+0.00)
	LMC-Memory	2.22 (-0.03)	**2.65 (-0.13)**	**3.21 (-0.03)**	0.02 (+0.00)	**0.03 (+0.00)**	**0.04 (+0.00)**

(c) Risultati quantitativi su KTH

Fig. 12: Risultati della estrapolazione DS e del guadagno/perdita (indicato tra le parentesi ottenuto con lo schema di estrapolazione ES [22].

Come si può osservare dalla Fig.11 e 12, ogni metodo, per poter essere adottato in maniera efficace, deve produrre una curva con valori che stiano al di sotto dei risultati ottenuti con Copylast. Questo per il motivo descritto precedentemente.
Osservando la Fig. 11 (a), nei risultati dei test effettuati con il dataset DriveSeg, i metodi SDCNet e FlowNet hanno $T_e > T_f$, il che influisce negativamente nel controllo di latenza al crescere dell'orizzonte temporale. Il metodo MCNet invece segue l'andamento desiderato, ma il tempo di estrapolazione comporta l'uso di un valore di

orizzonte temporale elevato, comportando una maggiore degradazione dell'immagine. Dalla Fig. 12 (a) si possono notare i risultati tramite le metriche BDPSNR e BDSSIM e il guadagno associato ad ogni risultato. Si osserva come effettivamente MCNet permetta una compensazione maggiore rispetto agli altri metodi, ma una degradazione maggiore della qualità ottenuta.
In contesti più semplici, sono stati ottenuti risultati accettabili tramite LMC-Memory e MCNet. Questo evidenzia l'importanza dei metodi veloci ed efficienti per la previsione video [22].

Il parametro T_e, ripetuto più volte in questo elaborato, ha un ruolo fondamentale nello studio della fattibilità della estrapolazione video. Come si può notare dalla formula (11), il tempo necessario alla estrapolazione influenza direttamente il valore della latenza G2G. Questa considerazione rende più realistici i risultati ricavati. È necessario che i metodi di estrapolazione siano associati alle tipologie di video più adatte a ciascuno di essi. Se il tempo di estrapolazione è maggiore al periodo di frame, la fattibilità della tecnica in analisi ha esito negativo. In futuro, per poter confrontare i risultati di studi differenti, è necessario che i test vengano eseguiti con dati il più realistici possibile. Questo si può ottenere soltanto considerando tutti gli elementi che influenzano la latenza in una normale trasmissione.

La sezione 4.4 copre una tematica differente dalle precedenti. Essa infatti, invece di testare differenti metodi di estrapolazione con vari dataset video, introduce un nuovo approccio di estrapolazione; ovvero combinare due tecniche, NVS e VP, per prevedere con maggiore precisione i frame futuri.
Nei risultati quantitativi è presente la metrica LPIPS (Learned Perceptual Image Patch Similarity). Essa cerca di fornire, mediante il confronto tra due frame, un valore che approssimi la percezione visiva umana.
I metodi confrontati sono delle stesse tipologie testate nelle altre sezioni.

Method	Extrapolation in Space			Extrapolation in Time		
	LPIPS↓	SSIM↑	PSNR↑	LPIPS-AlexNet↓		
				$t+1$	$t+3$	$t+5$
PredNet [41]		N/A		0.5535	0.5866	0.6295
MCNet [37]		N/A		0.2405	**0.3171**	**0.3739**
VoxelFlow [42]		N/A		0.3247	0.3743	0.4159
VEST [25]	**0.150**	**0.739**	**19.9**	**0.1560**	0.3441	0.4467

Fig. 13: Risultati ottenuti utilizzando dataset KITTI [33] [25].

I dati ottenuti dal confronto, nella Fig. 13, dimostrano come i metodi tradizionali possono risolvere solo un compito alla volta. Il metodo VEST [25] invece, fornisce sia

nello spazio che nel tempo valori differenti da N/A. Ciò permette un controllo maggiore sugli esiti dell'estrapolazione.

Un aspetto non approfondito adeguatamente è quello in merito alla complessità computazionale. Metodi come VEST e SDCNet, utilizzano approcci che richiedono una capacità di elaborazione elevata data la loro complessità. Questo richiede risorse, che non tutti i dispositivi dedicati alla trasmissione video garantiscono. È necessario quindi fare delle considerazioni finali.
Dai risultati quantitativi e qualitativi esaminati in questo elaborato, è stato introdotto e descritto il concetto di estrapolazione. Questa tecnica ha dimostrato, e continua a farlo, risultati promettenti per compensare la latenza nelle trasmissioni video. La versatilità dei metodi esaminati è però ancora scarsa. Il metodo che ha dimostrato una maggior capacità di adattamento, grazie all'approccio iterativo analizzato nella sezione 4.2, è SDCNet online.
È necessario che gli studi futuri approfondiscano le lacune annunciate precedentemente, standardizzando gli esperimenti. In questo modo sarà più semplice poter confrontare nuovi metodi con i precedenti in maniera pratica e veloce.

Bibliografia

[1] L. Pantel and L. C. Wolf, “On the impact of delay on real-time multiplayer games,” in Proceedings of the 12th International Workshop on Network and Operating Systems Support for Digital Audio and Video, New York, NY, USA, 2002, NOSSDAV ’02, p. 23–29, Association for Computing Machinery.

[2] S. Melnyk, A. Tesfay, K. Alam, H. Schotten, V. Sark, N. Maletic, M. Ramadan, M. Ehrig, T. Augustin, N. Franch, and G. Fettweis, “Reliable low latency wireless communication enabling industrial mobile control and safety applications,” 2018.

[3] S. K. Sharma, I. Woungang, A. Anpalagan, and S. Chatzinotas, “Toward tactile internet in beyond 5G Era: Recent advances, current issues, and future directions,” IEEE Access, vol. 8, pp. 56948–56991, 2020.

[4] H. Kanj, A. Trioux, M. Cagnazzo, F.X. Coudoux, P. Corlay, M. Kieffer, “On the feasability of efficient latency compensation using video frame extrapolation“

[5] M. Vijayaratnam, M. Cagnazzo, G. Valenzise, A. Trioux, and M. Kieffer, “Towards zero-latency video transmission through frame extrapolation,” in 2022 IEEE International Conference on Image Processing (ICIP), 2022, pp. 2122–2126.

[6] Z. Chang, X. Zhang, S. Wang, S. Ma, and W. Gao, “Strpm: A spatiotemporal residual predictive model for high-resolution video prediction,” 2022.

[7] R. Kreis, “Issues of spectral quality in clinical H-magnetic resonance spectroscopy and a gallery of artifacts”, NMR in Biomedecine, vol. 17, no. 6, pp. 361-381, 2004.

[8] I. Avcibas, B. Sankur and K. Sayood, “Statistical evaluation of image quality measures”, Journal of Electronic Imaging, vol. 11, no. 2, pp. 206-223, 2002

[9] J. E. Farrell, Image quality evaluation in colour imaging: vision and technology. MacDonald, L.W. and Luo, M.R. (Eds.), John Wiley, pp. 285-313, 1999.

[10] M. Cadik and P. Slavik, “Evaluation of two principal approaches to objective image quality assessment”, 8th International Conference on Information Visualisation, IEEE Computer Society Press, pp. 513-551, 2004.

[11] T. B. Nguyen and D. Ziou, “Contextual and non-contextual performance evaluation of edge detectors”, Pattern Recognition Letters, vol. 21, no.9, pp. 805-816, 2000.

[12] O. Elbadawy, M. R. El-Sakka, and M. S. Kamel, “An information theoretic image-quality measure”, Proceedings of the IEEE Canadian Conference on Electrical and Computer Engineering, vol. 1, pp. 169-172, 1998.

[13] A. Medda and V. DeBrunner, “Color image quality index based on the UIQI”, Proceedings of the IEEE Southwest Symposium on Image Analysis and Interpretation, pp. 213- 217, 2006.

[14] R. Dosselmann and X. D. Yang, “Existing and emerging image quality metrics”, Proceedings of the Canadian Conference on Electrical and Computer Engineering, pp. 1906-1913, 2006.

[15] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: from error visibility to structural similarity”, IEEE Transactions on Image Processing, vol. 13, no. 4, pp. 600-612, 2004.

[16] A. Horé, D. Ziou, “Image quality metrics: PSNR vs. SSIM”, 2010 International Conference on Pattern Recognition

[17] user: A. Alekhin, Git-Hub, “Similarity measurement (PSNR and SSIM)”, 2017, https://amroamroamro.github.io/mexopencv/opencv/image_similarity_demo.html

[18] F. A. Reda, G. Liu, K. J. Shih, R. Kirby, J. Barker, D. Tarjan, A. Tao, and B. Catanzaro, “SDC-Net: Video prediction using spatially-displaced convolution,” in Computer Vision – ECCV 2018, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds., Cham, 2018, pp. 747–763, Springer International Publishing.

[19] M. Vijayaratnam, M. Cagnazzo, G. Valenzise, A. Trioux, and M. Kieffer, “Towards zero-latency video transmission through frame extrapolation,” in 2022 IEEE International Conference on Image Processing (ICIP), 2022, pp. 2122–2126.

[20] “Recommendation itu-t p.910 - subjective video quality assessment methods for multimedia applications,” 2021.

[21] M. Vijayaratnam, M. Cagnazzo, G. Valenzise, E. Tartaglione, “All Predictions Matter: an Online Video Prediction Approach”, in 2023 IEEE 979-8-3503-4218-5/23

[22] M. Vijayaratnam, G. Valenzise, E. Tartaglione, M. Cagnazzo, “A latency compensation framework for video transmission based on frame extrapolation”, part of this work was presented at the IEEE Int. Conf. on Image Processing, Bordeaux, France, Oct. 2022

[23] L. Wang, S. Hong, and K. Panusopone, “Gradual decoding refresh for versatile video coding,” in 2021 IEEE International Conference on Image Processing (ICIP). IEEE, 2021, pp. 3448–3452

[24] X. Jiang, H. Shokri-Ghadikolaei, G. Fodor, E. Modiano, Z. Pang, M. Zorzi, and C. Fischione, “Low-latency networking: Where latency lurks and how to tame it,” Proceedings of the IEEE, vol. 107, no. 2, pp. 280–306, 2018.

[25] Y. Zhang, J. Wu, “Video Extrapolation in Space and Time”, arXiv:2205.02084v3 [cs.CV] 26 Jul 2022

[26]https://www.researchgate.net/figure/Example-images-and-ground-truth-annotations-in-the-Caltech-pedestrian-benchmark-Note_fig1_331102336

[27] H. Wang, X. Zhang, and al., “Inferring End-to-End Latency in Live Videos,” IEEE Trans. Broadcast., vol. 68, no. 2, pp. 517–529, 2022.

[28] C. Bachhuber, E. Steinbach, M. Freundl, and M. Reisslein, “On the minimization of glass-to-glass and glass-to-algorithm delay in video communication,” IEEE Transactions on Multimedia, vol. 20, no. 1, pp. 238–252, 2018.

[29] Zero-Lantency Linear Video Coding, https://zllvc.wp.imt.fr, finanziato dall'Agenzia Nazionale della Ricerca (ANR, Francia).

[30] P. Dollar, C. Wojek, B. Schiele, and P. Perona, “Pedestrian ´ detection: A benchmark,” in 2009 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), 2009, pp. 304–311.

[31] E. Ilg, N. Mayer, T. Saikia, et al., “Flownet 2.0: Evolution of optical flow estimation with deep networks,” in IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 2462–2470

[32]L. Ding, J. Terwilliger, R. Sherony, et al., “MIT driveseg (manual) dataset for dynamic driving scene segmentation,” Tech. Rep., Technical report, Massachusetts Institute of Technology, 2020.

[33]C. Schuldt, I. Laptev, and B. Caputo, “Recognizing human actions: a local svm approach,” in Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004. IEEE, 2004, vol. 3, pp. 32–36.

[34]S. Lee, H. G. Kim, D. H. Choi, H.-I. Kim, and Y. M. Ro, “Video prediction recalling long-term motion context via memory alignment learning,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 3054–3063.

[35] “Recommendation itu-t p.910 - subjective video quality assessment methods for multimedia applications,” 2021

[36] K. Soomro, A. R. Zamir, and M. Shah, “Ucf101: A dataset of 101 human actions classes from videos in the wild,” arXiv preprint arXiv:1212.0402, 2012.

[37] R. Villegas, J. Yang, S. Hong, et al., “Decomposing motion and content for natural video sequence prediction,” arXiv preprint arXiv:1706.08033, 2017.

[38] F. A. Reda, G. Liu, K. J. Shih, R. Kirby, J. Barker, D. Tarjan, A. Tao, and B. Catanzaro, “Sdc-net: Video prediction using spatially-displaced convolution,” in Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 718–733.

[39] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black, “A naturalistic open source movie for optical flow evaluation,” in Computer Vision– ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part VI 12. Springer, 2012, pp. 611–625.

[40] S. Lee, H. G. Kim, D. H. Choi, H.-I. Kim, and Y. M. Ro, “Video prediction recalling long-term motion context via memory alignment learning,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 3054–3063.

[41] Lotter, W., Kreiman, G., Cox, D.: Deep predictive coding networks for video prediction and unsupervised learning. In: ICLR (2017).

[42] Liu, Z., Yeh, R., Tang, X., Liu, Y., Agarwala, A.: Video frame synthesis using deep voxel flow. In: ICCV (2017).