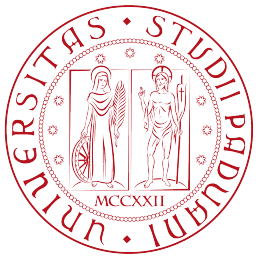

Università degli studi di Padova
Dipartimento di Ingegneria dell'Informazione
Corso di Laurea Triennale in Ingegneria dell'Informazione

Analisi degli standard di codifica per sequenze video 4K e HDR

Relatore
Prof. Simone Milani

Laureando
Alex Valente
matr. 1099466

Anno Accademico 2016/2017

Ai miei genitori e a mio fratello,
che mi hanno sempre sostenuto
Al mio relatore,
per la sua disponibilità e gentilezza
Ai miei più cari amici,
per il loro supporto

Indice

Elenco delle figure

Elenco delle tabelle

Sommario

La fruizione di contenuti multimediali in forma digitale è un aspetto quotidiano della nostra vita. Tra di essi i video svolgono un ruolo molto importante. Tuttavia la crescente richiesta di una maggiore qualità di visualizzazione e una distribuzione dei contenuti sempre più efficiente portano ad un aumento dei requisiti in termini di prestazioni. Pertanto, le tecniche di compressione video risultano in continua evoluzione. Questa tesi ha l'obiettivo di presentare la teoria di base su cui si fonda la codifica video, per poi descrivere gli standard sviluppati fino ad oggi. Infine si concentra sul confronto tra gli standard di codifica video più recenti, H.264 / AVC e H.265 / HEVC. I risultati sperimentali mostrano come quest'ultimo permetta di ottenere un notevole miglioramento dell'efficienza di codifica, in particolar modo riducendo la dimensione del bit stream codificato fino al 50%.

1

Introduzione

Negli ultimi decenni abbiamo assistito ad un'evoluzione tecnologica senza precedenti in svariati campi, sia per entità del cambiamento, sia per rapidità. Quest'ultimo fattore è stato il più incisivo, differenziando il panorama tecnologico in modo netto rispetto al passato.

In poco tempo siamo passati da un mondo multimediale prettamente analogico ad un altro in cui tutti i contenuti sono disponibili in formato digitale. Grazie alla digitalizzazione ed alla nascita di Internet abbiamo la possibilità di accedere in ogni momento ad una quantità infinita di informazioni, sia testuali che audiovisive. A differenza di venti anni fa, oggigiorno è comune la condivisione di immagini e video. Hanno certamente contribuito a questo la nascita dei social network, dei siti di video sharing (per es. YouTube) e la diffusione di dispositivi sempre più versatili e compatti come smartphone e tablet, i quali hanno cambiato le modalità di fruizione dei contenuti. Negli ultimi anni si è inoltre sviluppata la richiesta di maggiori qualità visiva e immersività, il tutto combinato con l'esigenza di diminuire la quantità di bit da memorizzare o trasmettere.

Ed è proprio per venire incontro a queste crescenti necessità che è stato di fondamentale importanza lo sviluppo di nuove e sempre più efficienti tecniche di rappresentazione dei contenuti multimediali.

Scopo di questo elaborato è dapprima la presentazione del funzionamento delle codifiche video, chiarendo la motivazione che ha spinto alla loro adozione. Verranno poi illustrate le peculiarità degli standard di compressione nati negli ultimi decenni per concentrarsi infine sull'analisi delle prestazioni qualitative e quantitative degli standard più recenti, ovvero H.264 / AVC e H.265 / HEVC, in una situazione che diverrà presto comune: la codifica di una sequenza video 4K e HDR.

2

Codifiche video

Questo capitolo è dedicato all'introduzione del concetto di codifica video e alle sue applicazioni.

2.1 Cos'è una codifica video

La compressione è definita come il processo con cui un determinato insieme di dati viene rappresentato in modo efficiente utilizzando una quantità ridotta di informazioni. La codifica video consiste quindi nel processo di conversione di un video digitale in uno stream binario di dimensioni più adatte alla trasmissione e all'immagazzinamento su memorie di massa.

Successivamente i termini "codifica" e "compressione" verranno utilizzati in modo ambivalente per fare riferimento allo stesso concetto qui presentato.

2.2 Perché usare codifiche video

Ogni dato acquisito o documento occupa un determinato spazio in memoria o su disco, misurato in bit. I contenuti multimediali quali immagini e video richiedono uno spazio di memorizzazione più elevato rispetto ad altri tipi di informazione. Ciò da sempre ha rappresentato un problema al quale, nel corso degli ultimi anni, sono state date diverse risposte. Ad oggi possiamo notare come sia sempre crescente la quantità di informazione che è possibile immagazzinare su supporti digitali mobili (ad oggi un singolo disco rigido è in grado di conservare una decina di TeraByte di dati). D'altra parte non si deve trascurare il fatto che questi contenuti, oltre ad essere memorizzati, devono anche essere trasmessi. La quantità di dati che possono essere inviati e ricevuti in un secondo è oggi enormemente più elevata rispetto ad anni fa. Ad esempio, le reti locali consentono oggi velocità di trasferimento dell'ordine del Gigabit per secondo. Anche le reti Internet, sia cablate sia mobili, hanno conosciuto negli ultimi anni un aumento notevole della capacità di throughput (si pensi allo sviluppo delle nuove tecnologie di accesso 4G e 5G).

Tuttavia, nonostante i progressi appena citati, i supporti di memorizzazione e le reti risultano essere comunque un collo di bottiglia in molteplici situazioni. Per esempio, un video 4K non compresso richiede circa 15 GB di spazio di archiviazione (vedi capitolo 3). Potrà sembrare una quantità accettabile date le caratteristiche (risoluzione 4K, pro-

fondità di pixel HDR, 60 fps), ma non bisogna dimenticare che parliamo di un video che ha una durata di soli 10 secondi. Se si considerasse un film con le stesse caratteristiche ma della durata di 2 ore, esso richiederebbe ben 10,8 TB, ovvero più della quantità di spazio che è in grado di memorizzare uno dei più recenti e capienti hard disk. Inoltre, dopo alcuni semplici calcoli si ottiene che il bitrate è di circa 12 Gbit/s. Nessun dispositivo oggi in commercio è in grado di trasferire dati ad una velocità simile (nemmeno i più recenti e veloci SSD), perciò il video non può essere riprodotto se non frame per frame. In aggiunta, è facile stimare che la quantità di dati da trasmettere non consente lo streaming della sequenza video attraverso una rete, locale o remota che sia.

La soluzione a tutti questi problemi è una sola: la compressione (o codifica). Essa consente di ridurre enormemente la quantità di dati necessaria per rappresentare lo stesso contenuto, a volte senza perdita di informazione (codifica lossless), altre volte con una perdita più o meno percettualmente rilevante (codifica lossy). È solo grazie alla nascita delle tecniche di compressione che oggi siamo in grado di acquisire, conservare e condividere foto e video con chiunque in modo semplice e veloce.

2.3 Acquisizione di una sequenza video

Qualunque scena vista dal nostro occhio si può considerare, matematicamente parlando, una proiezione di valori di luminosità rappresentabili da numeri reali su di un piano cartesiano in diversi istanti. Modellando il segnale come una funzione, possiamo pertanto dire che sia il dominio (dal punto di vista spaziale e temporale) sia il codominio sono continui. Per acquisire un'immagine 2D dal mondo reale lenti circolari focalizzano la radiazione luminosa proveniente dagli oggetti su sensori costituiti da un array di CCD (Charge Coupled Device) o CMOS (Complementary Metal-Oxide Semiconductor). Essi convertono quindi i segnali luminosi in segnali elettrici, che vengono poi elaborati. La rappresentazione in forma digitale comporta una discretizzazione dei dati.

2.3.1 Discretizzazione spaziale

Considerando la proiezione della scena su di un'immagine (es. un fotogramma di un video), essa viene partizionata in blocchi regolari, applicando cioè una griglia quadrata o rettangolare all'immagine acquisita. Tale griglia corrisponde alla matrice di sensori utilizzata dal dispositivo e viene misurato il valore di tensione. Ogni valore costituisce un pixel. Il numero di campioni che viene prelevato dall'immagine influenza pesantemente il livello di dettaglio che si otterrà alla fine del procedimento.

2.3.2 Discretizzazione temporale

Il campionamento dal punto di vista del tempo consiste nel catturare un'immagine ad intervalli regolari nel tempo. Una frequenza di campionamento elevata porta ad avere

un video in cui i movimenti sono fluidi e molto simili a quelli naturali. Al contrario, una frequenza bassa porta ad avere una visione "a scatti". Valori tipici sono dell'ordine dei 25-30 frame per secondo per video a definizione standard, 50-60 frame per secondo quando si considera l'alta definizione. Frequenze di 10 fps sono usate solo per applicazioni quali le videoconferenze o le videochiamate tramite rete mobile, in quanto un valore così basso permette di ridurre la quantità di dati da trasmettere, garantendo così una comunicazione in tempo reale seppur a discapito della qualità.

2.3.3 Profondità di colore o bit depth

Il valore di ogni pixel è rappresentato da un numero intero a precisione finita rappresentato da un certo numero di bit. La profondità di colore adottata è individuata dal numero di bit di cui è costituita ogni rappresentazione. Tipicamente si utilizzano 8 bit, ma sempre più diffuso è l'uso di 10 bit. In questo caso si parla di immagine HDR (High Dynamic Range).

2.3.4 Altri dettagli

Quello che è stato descritto finora è chiamato campionamento di tipo progressivo. Grazie ad esso ogni frame risulta costituito da tutti i pixel di cui è formato. Un altro tipo di campionamento molto diffuso è invece quello interlacciato. Esso richiede la suddivisione di ogni frame in linee orizzontali. In una sequenza video interlacciata ogni fotogramma è costituito dalle sole linee con indice pari o dispari, con la conseguenza che la risoluzione in altezza risulta dimezzata. Il vantaggio di questo approccio consiste nella possibilità di trasmettere una quantità doppia di frame rispetto al progressivo, mantenendo però lo stesso bitrate. Il ricevente ha così una visione più fluida del filmato senza una perdita di qualità percettualmente rilevante. Il passaggio da una tecnica di campionamento ad un'altra comporta un'operazione di conversione che richiede un'elaborazione aggiuntiva e la possibile presenza di artefatti. Poichè la nascita del campionamento interlacciato è strettamente legato al funzionamento degli schermi analogici del passato, esso sta rapidamente lasciando il passo a quello progressivo.

2.4 Funzionamento di una generica codifica video

I diversi standard di codifica video che sono stati sviluppati negli ultimi vent'anni condividono lo stesso schema di base. Ciò che li distingue è infatti principalmente l'efficienza, oltre all'applicazione di destinazione per cui sono stati pensati.

Per realizzare un sistema di codifica sono necessari due componenti tra loro complementari, ovvero un encoder (codificatore, compressore) ed un decoder (decodificatore, decompressore). L'encoder si occupa di convertire i dati della sorgente in un bit stream, richiedendo la minima quantità possibile di bit per la sua rappresentazione. Questo

va effettuato prima della memorizzazione e dell'eventuale trasmissione. Per far ciò si rimuove la ridondanza statistica insita nei dati, applicando una cosiddetta codifica di sorgente. La codifica può essere senza perdita di informazione (codifica lossless): essa consente di ricostruire il dato originale a partire dal bit stream codificato in modo fedele senza alterazioni. Per ottenere risultati più efficienti dal punto di vista della dimensione del bit stream, si accetta spesso di avere una perdita più o meno significativa (codifica lossy).

Il decoder invece è necessario per riconvertire i dati compressi nella loro forma originaria, o approssimarla il più possibile se si è optato per una codifica lossy.

La coppia enCOder/DECoder è indicata con la parola codec.

Nei paragrafi successivi verranno introdotti i concetti base riguardanti la rappresentazione delle immagini, per poi concentrarsi sulla sequenza di operazioni che portano ad ottenere uno stream codificato a partire da una sequenza di frame video non compressi.

2.4.1 Rappresentazione dei colori

Per la rappresentazione di un'immagine è necessario codificare per ogni pixel informazioni quali la luminosità e i colori. Esistono varie rappresentazioni chiamate spazi colore.

RGB

Nello spazio colore RGB, ogni pixel è rappresentato da 3 numeri interi che indicano la quantità di rosso, verde e blu. Combinando questi tre colori si può creare ogni altro colore. Ogni immagine è quindi rappresentata da 3 matrici, una per ogni componente di colore.

Come già accennato, il numero di bit utilizzato per rappresentare una singola componente determina la profondità di colore dell'immagine e quindi l'effettivo numero di tonalità che possono essere rappresentate. L'utilizzo di 8 bit per componente consente di avere $2^8 = 256$ possibili tonalità di rosso, verde e blu. Combinandole insieme il numero di colori rappresentabili è $(2^8)^3 = 2^{24} = 16,8$ milioni circa. Grazie all'utilizzo di 10 bit per componente il numero di colori diventa $(2^{10})^3 = 2^{30} = 1,07$ miliardi.

YCbCr

Lo spazio colore YCbCr deve la sua origine ad una considerazione riguardo il sistema visivo dell'uomo (HVS = Human Visual System). Infatti è noto come l'occhio umano sia meno sensibile ai colori rispetto alla luminosità. RGB non tiene conto di questa osservazione e quindi dà la stessa importanza ad ognuna delle 3 componenti. YCbCr invece separa le informazioni relative alla luminosità (luminanza, Y) da quelle riguardanti i colori (crominanza blu Cb, crominanza rossa Cr e crominanza verde Cg). Si definisce quindi la luminanza Y come media pesata tra le componenti dei colori:

$$Y = k_r * R + k_g * G + k_b * B$$

Le crominanze rossa Cr, blu Cb e verde Cg sono invece la differenza tra l'intensità della componente colore e Y:

$$Cr = R - Y \qquad Cb = B - Y \qquad Cg = G - Y$$

Si potrebbe pensare che questa rappresentazione sia svantaggiosa dato che prevede 4 componenti contro le 3 di RGB. Tuttavia si può notare che la somma tra le 3 crominanze è una costante, ovvero proprio la luminanza Y. Perciò, per esempio, il valore di Cg si può ricavare come differenza tra Y e le altre 2 crominanze debitamente pesate. Non è quindi necessario che venga trasmessa.

Inoltre, vista l'osservazione riguardante l'HVS, è possibile rappresentare le componenti Cb e Cr con una risoluzione inferiore rispetto ad Y. Gli schemi di decimazione delle crominanze (*chroma subsampling*) più frequentemente usati sono:

- 4:4:4

  Ognuna delle 3 componenti ha la stessa risoluzione e quindi ad un campione di Y corrisponde un campione Cb ed uno Cr.

- 4:2:2

  Le componenti di crominanza hanno la stessa risoluzione verticale della componente di luminanza, ma la risoluzione orizzontale è invece dimezzata. Perciò ogni 4 campioni di Y ce ne sono 2 di Cb e 2 di Cr.

- 4:2:0

  La risoluzione delle componenti di crominanza è dimezzata sia verticalmente che orizzontalmente e quindi a 4 campioni Y corrispondono un solo campione Cb ed uno Cr. È il formato più diffuso nelle applicazioni consumer quali TV, DVD e videoconferenze.

Come per RGB ogni immagine è rappresentata da 3 matrici, sebbene abbiano un significato diverso. La differenza fondamentale sta nel fatto che esse possono non avere la stessa dimensione. L'utilizzo dello spazio colore YCbCr può quindi essere considerato una prima forma di compressione.

2.4.2 Predizione

Una sequenza video è formata da immagini che vengono visualizzate una dopo l'altra in modo da rappresentare, almeno nel caso di filmati dinamici, soggetti in movimento. Tra un fotogramma e l'altro però solitamente non tutta l'immagine varia. Si può perciò sfruttare la correlazione tra frame successivi per ridurre la quantità di informazioni da memorizzare.

Anche all'interno di un singolo fotogramma è possibile trovare una correlazione tra pixel adiacenti. Un esempio è dato da un'immagine che rappresenta una scena naturale,

in cui una porzione è occupata dal cielo. In questa porzione la variazione locale del colore è poco visibile e rappresenta quindi un candidato ideale per una predizione spaziale.

La predizione di un frame basata sui pixel vicini tra loro (*Intra-frame*) e sui fotogrammi temporalmente adiacenti (*Inter-frame*) è un elemento fondante di tutte le codifiche video e le tecniche di predizione sono state continuamente migliorate nel passaggio da uno standard al successivo.

Predizione spaziale Intra-Frame

Il fotogramma viene suddiviso in unità fondamentali chiamate blocchi. I blocchi sono di forma quadrata e sono costituiti da più pixel. Dimensioni tipiche sono 4x4 pixel, 8x8 o 16x16 (nella codifica H.264 / AVC) e la dimensione può essere differente per componenti distinte dell'immagine (per esempio con lo spazio colore YCbCr si possono avere blocchi di dimensione 16x16 pixel per la luminanza e 8x8 pixel per le componenti di crominanza). Più la dimensione è piccola, maggiore è l'accuratezza della predizione e minore è il numero di bit richiesti per codificare il blocco. Tuttavia è necessario un certo numero di bit per segnalare al decodificatore che un certo blocco non è rappresentativo dell'immagine originale ma è invece il risultato di una predizione e per specificare tutte le informazioni necessarie a riprodurre la predizione. Perciò potrebbe accadere che la quantità di bit risparmiati non trasmettendo le informazioni sui pixel risulti inferiore a quella necessaria per le segnalazioni. Ciò porta quindi a considerare l'idea di utilizzare blocchi più grandi. In sintesi, la predizione intra-frame utilizza i campioni dei blocchi adiacenti già codificati per predire i valori delle componenti di colore o di luminanza e crominanza del blocco considerato.

Le tecniche di compressione più avanzate come la recente H.265 / HEVC fa uso anche di macro-blocchi con dimensioni di 64x64 pixel, ma allo stesso tempo può anche utilizzare partizioni più complesse rispetto a quelle quadrate.

Predizione temporale Inter-Frame

La predizione inter-frame è nota anche come stima del moto. Così come la predizione intra-frame, essa si basa sulla suddivisione di ogni immagine in blocchi o macro-blocchi formati da pixel. Una volta considerato un fotogramma, se ne prende un altro già codificato e quest'ultimo diventa il frame di riferimento (*reference frame*). Per ogni blocco del frame considerato, si cerca nel reference frame il blocco che più lo approssima. Per ottimizzare la ricerca ci si può limitare a cercare il blocco candidato in un'area limitata attorno al blocco considerato. Quando il candidato migliore è stato trovato, si crea un vettore di movimento (*motion vector*) che esprime l'offset orizzontale e verticale tra i blocchi dei due fotogrammi. L'operazione viene quindi eseguita per ogni blocco e si ottiene infine un insieme di vettori che indicano l'eventuale spostamento dei blocchi. Dalla collezione di vettori si ricavano i blocchi che vengono predetti. Infine ogni blocco predetto viene sottratto al blocco di riferimento per ottenere un blocco residuo, quello che effettivamente verrà memorizzato o trasmesso. Anche in questo caso minore è la dimen-

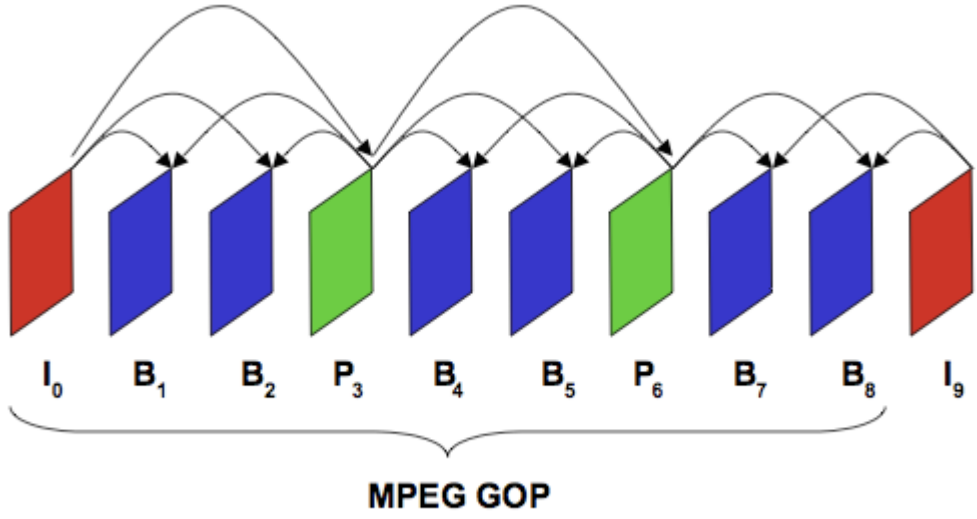

Figura 2.1.: Esempio di *Group of Pictures*

sione dei blocchi maggiore è l'attendibilità della predizione, a discapito della capacità di compressione.

Una sequenza video può essere considerata formata da gruppi di immagini detti GOP (*Group Of Pictures*). La struttura di un GOP determina quale o quali sono i reference frame per un determinato fotogramma. Un GOP può contenere frame di diversi tipi:

- **I-frame** (Intra-coded frame): consiste di un'immagine che viene compressa mediante la sola predizione spaziale, senza fare riferimento a nessun altro frame del GOP. Fa anzi da riferimento per tutti gli altri tipi di fotogrammi, essendo quello che approssima certamente con più fedeltà l'immagine originale. Un Group Of Pictures contiene sempre almeno un I-frame.

- **P-frame** (Predictive frame): utilizza come reference frame un frame precedente. Richiede meno bit rispetto ad un I-frame dato che vengono codificate solo le differenze tra il frame considerato e quello di riferimento (che può essere sia un I, un B o un altro P).

- **B-frame** (Bi-predictive frame): utilizza come reference frame sia un fotogramma precedente che uno successivo. La compressione che si ottiene è superiore rispetto a quella dei frame I e P, ma soffre di una latenza maggiore dato che per la codifica è necessario attendere che sia stato compresso il frame temporalmente successivo. Ciò provoca quindi una codifica out-of-order.

Sebbene sia ancora largamente utilizzata la predizione a blocchi e macroblocchi, numerosi studi hanno dimostrato che il partizionamento orientato agli oggetti (ovvero adattando le regioni alle dimensioni degli oggetti rappresentati in una sequenza video) porta a risultati sensibilmente migliori, a discapito però della crescente complessità computazionale che

ciò comporta. HEVC è il primo standard di codifica ad adottare un approccio ibrido tra quello a blocchi e quello ad oggetti.

2.4.3 Codifica a trasformate

Ora che sono stati definiti quali sono e come sono strutturati i frame che andranno a comporre la sequenza video, si passa alla fase di vera e propria codifica lossy dei singoli fotogrammi. Essa è composta da numerose fasi, ovvero:

- Preparazione dei blocchi
- Passaggio al dominio della trasformata
- Quantizzazione
- Codifica entropica

Preparazione dei blocchi

Operando la stessa suddivisione in blocchi e macro-blocchi utilizzata per la predizione, ogni valore dei pixel di cui è composto viene centrato in 0 sottraendo $2^{(n-1)}$, dove n è il numero di bit che viene usato per la rappresentazione. Quindi nel caso di profondità di colore a 8 bit si sottrae 128, nel caso di 10 bit (HDR) 512. Questa operazione va effettuata su ogni componente colore se lo spazio colore utilizzato è RGB, solo su Y se è YCbCr.

Passaggio al dominio della trasformata

Questa operazione permette di ridurre la correlazione, così da concentrare in un numero minore possibile di valori l'energia del segnale considerato. La trasformata deve essere reversibile e computazionalmente implementabile, senza richiedere un utilizzo di risorse eccessivo. Come per la compressione delle immagini JPEG, anche in MPEG-1 e successivi si predilige l'uso della trasformata DCT (*Discrete Cosine Transform*) invece della classica FFT (*Fast Fourier Transform*) poiché raggiunge risultati sensibilmente migliori nella riduzione della ridondanza, a discapito però di una complessità computazionale maggiore. La trasformata discreta del coseno (DCT) opera su $\mathbf{X}$, un blocco di N x N campioni visto come matrice, fatti da pixel ottenuti direttamente dall'immagine originale o tramite predizione. Ciò che calcola è $\mathbf{Y}$, ovvero un blocco delle stesse dimensioni di $\mathbf{X}$ ma fatto da coefficienti. $\mathbf{X}$ ed $\mathbf{Y}$ sono legati dalle seguenti relazioni matriciali:

$$\mathbf{Y} = \mathbf{A}\mathbf{X}\mathbf{A}^T \qquad \mathbf{X} = \mathbf{A}^T\mathbf{Y}\mathbf{A}$$

$\mathbf{A}$ è la matrice N x N per la trasformata e i suoi valori sono dati da:

$$A_{ij} = C_i \cos\frac{(2j+1)i\pi}{2N} \qquad dove \quad C_i = \begin{cases} \sqrt{\frac{1}{N}} & \text{se } i = 0 \\ \sqrt{\frac{2}{N}} & \text{se } i > 0 \end{cases}$$

Ogni coefficiente dato in output dalla trasformata DCT è una combinazione lineare tra un set predeterminato di N x N pattern. Quello che si ottiene infine è una nuova matrice nella quale è possibile identificare alcune sezioni significative.

- Il primo valore in alto a sinistra ($\mathbf{Y}(0,0)$) è il coefficiente DC. Esso rappresenta la componente continua (a frequenza nulla), ovvero il colore di base dell'intero blocco.
- La prima riga (escluso il primo valore) è data dai coefficienti AC orizzontali.
- La prima colonna (escluso il primo valore) è data dai coefficienti AC verticali.
- Tutti gli altri valori sono anch'essi coefficienti AC e rappresentano la variazione della componente R, G, B o Y rispetto al coefficiente DC.

Allontanandosi dalla componente DC la componente AC considerata rappresenta una frequenza sempre maggiore.

Nota la differenza tra i coefficienti AC e DC, si può concludere che se il blocco considerato è costituito prevalentemente da un singolo colore allora la matrice avrà pochi coefficienti AC non nulli, mentre accadrà esattamente l'opposto nel caso in cui l'immagine contenga molte sfumature di colore.

Quantizzazione

La quantizzazione consiste nel mappare valori presi da un intervallo continuo in altri valori distribuiti su di un intervallo limitato e discreto. Questo intervallo viene diviso in livelli di ampiezza uguale (quantizzazione uniforme) oppure differente (quantizzazione non uniforme). Di fondamentale importanza è il numero di livelli in cui questo intervallo viene suddiviso e di conseguenza la loro dimensione, chiamata passo di quantizzazione. La quantizzazione è infatti un processo non invertibile che porta sempre ad una perdita di informazione (compressione lossy), la cui entità dipende esclusivamente dal numero di livelli che vengono considerati. Ciò infatti più di qualsiasi altro parametro determina la qualità visiva che si ottiene come output della codifica.

Nel caso in esame la trasformata DCT fornisce già coefficienti interi, tuttavia è possibile ugualmente ridurre il numero di valori utilizzati, facendo per esempio tendere a zero i coefficienti con un valore poco significativo. Come fatto per YCbCr, per ottenere una buona quantizzazione si possono fare osservazioni riguardanti l'HVS (Human Visual System). L'occhio umano difatti è più sensibile alle basse frequenze rispetto a quelle alte. Motivo per cui si adotta una quantizzazione fine (si usano cioè più bit per campione) per la componente DC e le componenti AC a bassa frequenza, una quantizzazione più grossolana (meno bit per campione) per le componenti AC ad alta frequenza. Questa differenza è indicata nella matrice di quantizzazione con il valore di soglia (o passo) che aumenta allontanandosi dal valore in alto a sinistra. Dividendo ogni coefficiente della DCT per il corrispondente valore di soglia e arrotondando il risultato si ottiene la nuova matrice dei coefficienti quantizzati.

10	10	15	20	25	30	35	40
10	15	20	25	30	35	40	50
15	20	25	30	35	40	50	60
20	25	30	35	40	50	60	70
25	30	35	40	50	60	70	80
30	35	40	50	60	70	80	90
35	40	50	60	70	80	90	100
40	50	60	70	80	90	100	110

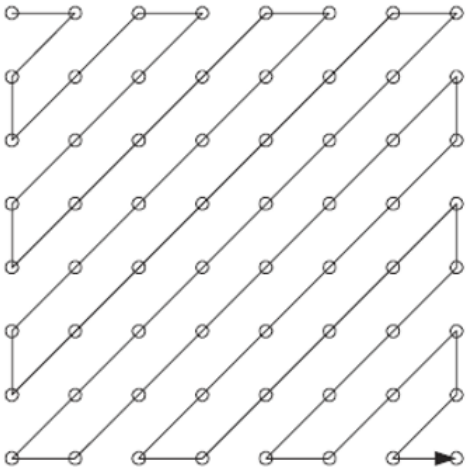

Figura 2.2.: Esempio di matrice di quantizzazione e ordine di lettura per Zig-Zag Scanning

Codifica entropica

Quest'ultimo passo porta al passaggio dalla struttura matriciale dei frame ad ottenere una sequenza di bit più corta possibile che li rappresenti. Si compone di 4 sotto-passi:

1. **Vectoring**

   Tutti gli algoritmi di codifica entropica operano solo su vettori monodimensionali e non su matrici. Pertanto si utilizza la tecnica dello Zig-Zag scanning per leggere i valori dei coefficienti DCT in ordine tale da riportarli in un vettore e da concentrare i coefficienti diversi da zero nelle prime posizioni lasciando, allo stesso tempo, i coefficienti nulli nelle ultime. Questo ordine rende alcune tecniche di codifica entropica successive (come, ad esempio, la codifica Run-Length) particolarmente efficaci.

2. **Differential encoding (per coefficienti DC)**

   Poiché blocchi adiacenti hanno coefficienti DC simili, è sufficiente usare il primo valore come riferimento e codificare gli altri con la differenza rispetto al primo. Per codificare la differenza si usa il formato (SSS, value), dove SSS indica il numero di bit di cui è composto il valore, mentre value è il valore stesso in binario.

3. **Run-Length encoding (per coefficienti AC)**

   Questa codifica è indicata per l'uso con i coefficienti delle componenti ad alta frequenza perché contengono lunghe sequenze di bit uguali a 0. Il formato usato per la codifica è (skip, value), dove skip è il numero di zeri da saltare prima di arrivare a value. Quando si incontra la codifica (0,0) vuol dire che tutti i rimanenti bit sono 0. Anche value necessita però di essere codificato. Per esso si utilizza il formato SSS/value, il cui significato è lo stesso della codifica differenziale.

4. **Codifica di Huffman**

   È utilizzata per rappresentare il campo SSS della codifica differenziale e le coppie (skip, SSS) della codifica Run-Length. La codifica di Huffman utilizzata è una codifica binaria a prefisso e a lunghezza variabile: le parole più frequenti sono codificate con parole di codice più corte, quelle più rare utilizzano invece un numero maggiore di bit per la loro rappresentazione.

La codifica di Huffman non è l'unica possibile. Essa può essere infatti sostituita dalla codifica aritmetica.

2.5 Esempi di codifiche video

Per la codifica di contenuti video negli ultimi decenni si sono susseguiti numerosi standard. Essi sono stati proposti da 2 organizzazioni riconosciute a livello mondiale:

- **ITU-T** è uno dei tre settori di cui è composto ITU (International Telecommunication Union). ITU nacque nel lontano 1865, per poi diventare un'agenzia specializzata facente parte delle Nazioni Unite nel 1947. Il nome odierno ITU-T (la cui ultima parte del nome sta per Telecommunication Standardization Bureau) è abbastanza recente, essendo datato 1993. Lo scopo di questa organizzazione è stato ed è tutt'ora quello di produrre standard che fossero validi a livello mondiale nel campo delle telecomunicazioni, le cosiddette "Raccomandazioni". Quelle che riguardano i sistemi audiovisivi e multimediali, oggetto di questa tesi, sono quelle della serie H (es. H.264, H.265...).

- **ISO/IEC** (acronimo di International Organization for Standardization / International Electrotechnical Commission), il cui comitato di lavoro MPEG (Moving Picture Experts Group, anche noto in modo più formale come ISO/IEC JTC 1/SC 29/WG 11) dal 1988 si occupa fondamentalmente di realizzare standard per la compressione e la trasmissione di audio e video.

Queste due organizzazioni sono indipendenti tra loro. Mentre la prima è nota per aver dato alla luce H.261 e H.263 (quest'ultimo molto utilizzato per le videoconferenze), la seconda è conosciuta per aver prodotto MPEG-1 e MPEG-4 Visual (questo purtroppo rivelatosi un insuccesso vista la scarsissima diffusione sebbene inizialmente fosse molto promettente, come specificato in [1]). Tuttavia alcuni degli standard oggi più popolari e di evidente successo sono nati dalla stretta collaborazione tra esse. Parliamo infatti di H.262 / MPEG-2 Video e H.264 / MPEG-4 Advanced Video Coding (AVC), oltre al nuovissimo H.265 / HEVC.

2.5.1 MPEG-1

MPEG-1 è uno standard sviluppato da MPEG tra il 1988 e il 1992 con lo scopo di comprimere in modo lossy sequenze video e audio. Benchè tecnicamente in grado di codificare video con formati anche superiori all'odierno 4K, esso venne ottimizzato per applicazioni video con basse risoluzioni ed un bitrate di 1.5 Mbit/s, in modo da ottenere una significativa compressione delle immagini ed una qualità simile a quella delle cassette VHS (Video Home System). Il codec MPEG-1 venne largamente utilizzato nei video CD e inizialmente nei DVD, anche se presto fu sostituito dal più performante MPEG-2. Un grande successo lo ebbe piuttosto il codec audio facente parte dello standard, ovvero MPEG-1 Layer III, noto come MP3.

Dal punto di vista video, il funzionamento della codifica rispecchia quasi fedelmente la procedura classica presentata nel capitolo precedente. L'unica differenza degna di nota è la presenza, scomparsa negli standard video successivi, dei D-frame, oltre ai già conosciuti I, P e B-frame (vedi [4]). Essi consistono in immagini indipendenti dalle altre, la cui codifica è di tipo intra-frame e fa uso dei soli coefficienti DC della trasformata, rimuovendo quindi completamente gli AC. Di conseguenza questi frame, tranne in casi molto rari come un'immagine consistente in una superficie di un solo colore, hanno una qualità nettamente inferiore rispetto agli altri. Vengono perciò utilizzati solo quando viene visualizzata un'anteprima ad alta velocità del video, per esempio durante la ricerca di una determinata scena. Non a caso questo ricorda l'avanzamento e il riavvolgimento del nastro di una cassetta VHS. Tuttavia nel caso in cui il decoder sia abbastanza veloce, i D-frame possono essere tranquillamente sostituiti dagli I-frame.

Alcuni limiti che hanno portato alla nascita degli standard successivi sono stati certamente la qualità modesta delle immagini, la compressione migliorabile quando si aumenta la risoluzione del video da codificare, la disponibilità del solo profilo colore 4:2:0 e il supporto a due soli canali audio. Non è da meno però un altro fatto, ovvero il mancato supporto per i video interlacciati, oggi in rapido abbandono ma allora molto diffusi. Ciò infatti richiedeva un doppio processo di codifica, con un costo in termini di tempo di computazione non trascurabile.

2.5.2 H.262 / MPEG-2

H.262, conosciuto anche come MPEG-2, è uno standard sviluppato congiuntamente da ITU-T e MPEG all'inizio degli anni '90 ed introdotto ufficialmente nel 1994. Condivide gran parte delle caratteristiche di MPEG-1, migliorandolo però nei suoi punti critici. La qualità video aumenta drasticamente grazie anche all'ottimizzazione per bitrate maggiori del conservativo 1.5 Mbit/s: si parla in questo caso di 4 - 9 Mbit/s. Aggiunge inoltre il supporto per i video interlacciati e per l'audio multicanale. Date le numerose applicazioni possibili, quali l'impiego nei DVD e nella TV digitale (DVB - Digital Video Broadcasting), ed essendo adattabile sia a video a definizione standard che a quelli in

alta definizione, di fondamentale importanza è stata l'introduzione dei profili e dei livelli. I profili consentono di definire un insieme limitato di caratteristiche che l'applicazione può utilizzare. I livelli invece permettono di definire parametri come il bitrate massimo dei video codificati, il frame rate e la loro dimensione, in modo da limitare la quantità di risorse di sistema richieste durante l'utilizzo del codec. La combinazione tra un profilo ed un livello consente all'utilizzatore del codec di avvalersi di una configurazione già ottimizzata per una determinata applicazione, senza dover impostare tutti i parametri manualmente.

Vista la somiglianza con il predecessore MPEG-1, MPEG-2 ha potuto diffondersi rapidamente in quanto il relativo codec, e di conseguenza il software e l'hardware in cui è implementato, è retrocompatibile con il vecchio standard. Oggi MPEG-2 è ancora molto diffuso, essendo l'unico codec in uso nella TV digitale DVB, sia per le trasmissioni a risoluzione standard che per quelle ad alta risoluzione. Questo poiché quello che avrebbe dovuto essere il successore ottimizzato per questo impiego, ovvero MPEG-3, non è mai stato rilasciato in quanto non portava miglioramenti degni di nota e divenne quindi un semplice nuovo profilo di MPEG-2 (chiamato MPEG-2+). Anche i filmati salvati su supporti ottici Blu-Ray Disc sono stati inizialmente codificati con MPEG-2, successivamente sostituito da H.264 / AVC.

2.5.3 H.264 / AVC

H.264 / Advanced Video Coding (AVC) (noto anche come MPEG-4 Recommendation 10) è uno dei più recenti standard di codifica video sviluppato grazie alla collaborazione tra ITU-T VCEG (Video Coding Experts Group) e ISO/IEC MPEG. Nato alla fine degli anni '90 ma pubblicato ufficialmente nel 2003, è stato esteso negli anni successivi per supportare nuovi formati video, tra cui la risoluzione 4K e HDR. Alcune soluzioni definite in H.264 / AVC sono di implementazione facoltativa, lasciando agli sviluppatori la scelta circa quali utilizzare e quali no. Ciò però può causare problemi di compatibilità tra diversi encoder e decoder. La soluzione arriva dalla definizione, come già anticipato in MPEG-2, di numerosi profili e livelli, i quali determinano quali soluzioni di codifica sono utilizzate e regolano di conseguenza le applicazioni di destinazione e la complessità computazionale.

Rispetto agli standard precedenti le prestazioni in termini di efficienza sono notevolmente migliorate, dando la possibilità di comprimere una sequenza video con un numero minore di bit a parità di risoluzione e qualità dell'immagine. Questo è dovuto in particolar modo alle modifiche che coinvolgono il sistema di predizione. Se prima un frame poteva essere codificato come di tipo I, P o B, ora esso può essere suddiviso in regioni o gruppi di macroblocchi 16x16 chiamate slice le quali possono avere codifiche diverse. Un singolo fotogramma può essere quindi composto da slice di tipo I, P, B, SP e SI (gli ultimi due sono switching slices).

Oltre ad aver sostituito in molte applicazioni il suo predecessore MPEG-2, esso ha

avuto molte applicazioni aggiuntive. È stato utilizzato infatti per le trasmissioni televisive principalmente in alta definizione veicolate tramite satellite, cavo e antenna. Inoltre viene usato in sistemi di acquisizione quali videocamere, smartphone e impianti di videosorveglianza. Non è da dimenticare l'uso che ne viene fatto per trasmettere video tramite la rete Internet, comprimere i film memorizzati nei supporti Blu-Ray, oppure nelle videoconferenze e nei sistemi di telepresenza. Si è rivelato dunque uno tra i sistemi di compressione più versatili in assoluto.

2.5.4 H.265 / HEVC

H.265 / HEVC è il nuovo standard di codifica video risultato del lavoro congiunto tra ITU-T VCEG e ISO/IEC MPEG, i quali hanno lavorato a questo progetto formando il JCT-VC (Joint Collaborative Team on Video Coding). La prima versione dello standard è datata gennaio 2013, mentre le successive due versioni sono rispettivamente nate nel 2014 e 2015 come estensione del codec originale. La nascita di HEVC non è dovuta solo alla volontà di aumentare la capacità di compressione (il proposito è quello di ottenere una compressione doppia rispetto ad AVC a parità di condizioni), ma anche a quella di consentire lo sviluppo di nuovi servizi, rendendo possibile la fruizione di contenuti in altissima risoluzione quali 4K e 8K, con una profondità di colore maggiore di quella classica (HDR) e con un livello di dettaglio mai visto prima.

HEVC richiede una complessità computazionale maggiore rispetto ad AVC, tuttavia è stato studiato per sfruttare l'hardware odierno permettendo la parallelizzazione di encoding e decoding. Ciò quindi porta ad annullare lo svantaggio nella maggioranza dei casi (oggi i processori multi-core sono diffusi in ogni tipo di piattaforma, sia essa un computer, uno smartphone o un tablet). Con la crescente diffusione di dispositivi mobili anche il consumo energetico è diventato di fondamentale importanza. Per far fronte a ciò sempre più spesso il supporto a HEVC viene inserito direttamente nell'hardware dei processori o delle GPU (Graphic Processing Unit) tramite sezioni di circuito appositamente studiate. Oltre a permettere prestazioni in termini di tempo migliori, questo contribuisce ad un minor consumo della batteria.

HEVC differisce da AVC sotto numerosi aspetti tecnici. Essendo studiato per codificare immagini ad altissima risoluzione, il limite di 16 x 16 pixel come grandezza massima dei blocchi risulta poco conveniente. È molto probabile trovare scene in cui la compressione potrebbe agire su aree più vaste, per esempio 64 x 64 pixel. Ogni immagine viene partizionata in CTB (*Coding Tree Block*) di forma quadrata, in numero uguale per le componenti di luminanza e crominanza se lo schema di decimazione di quest'ultima è 4:4:4. Se, come è più probabile, il chroma subsampling è 4:2:0, le CTB di crominanza hanno un'area che è 1/4 di quella della componente di luminanza. Ogni CTB ha dimensione L x L dove L viene scelto tra 16, 32 o 64 campioni. L'insieme fatto dalle tre CTB e dagli elementi di sintassi forma l'unità fondamentale CTU (*Coding Tree Unit*). Ogni CTU è però a sua volta divisibile in più CU (*Coding Units*) di dimensione variabile,

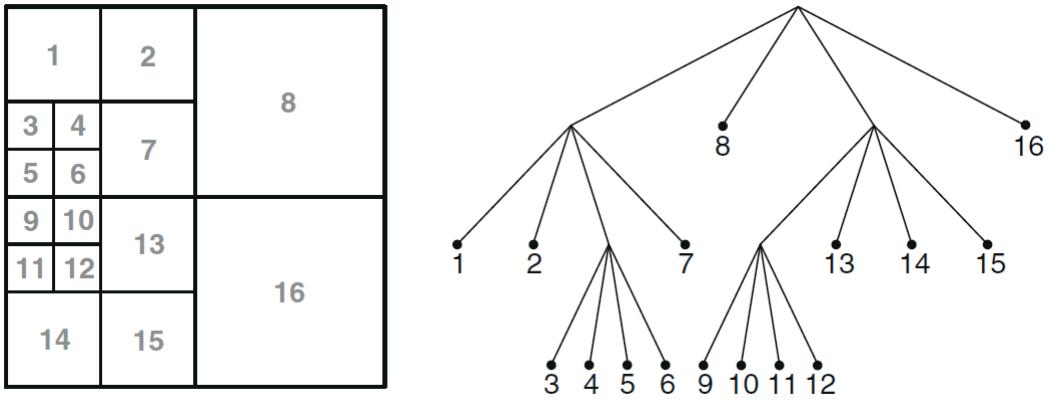

Figura 2.3.: Esempio di partizionamento di una CTU 64x64 in CU con dimensione da 8x8 a 32x32

consistenti in blocchi più piccoli per ogni componente chiamati CB (*Coding Blocks*). Da ciò deriva la possibilità di vedere un CTU sottoforma di albero quaternario (quad-tree), dove la radice rappresenta il blocco più grande, mentre ogni sottoalbero un CU, il quale può essere a sua volta suddiviso in ulteriori CU. È evidente quindi come la scelta di usare una struttura gerarchica sia l'ideale. Un esempio di questo partizionameto è visibile nella Figura 2.3. La scelta riguardante il tipo di predizione da effettuare (intra-frame o con compensazione del moto) viene fatta sulle CU.

- Se la predizione scelta è intra-frame, sono ben 35 le modalità tra cui è possibile scegliere: DC, planare oppure confrontando con un blocco raggiungibile spostandosi lungo una delle 33 direzioni possibili (per fare un confronto, in H.264 esse erano solo 8).

- Se invece si è optato per la predizione inter-frame, è possibile un'ulteriore suddivisione delle CB in PB (*Prediction Blocks*) con dimensione che può arrivare ad un minimo di 4 x 4. Esse costituiscono le PU (*Prediction Units*). Ogni PB ha uno o più MV (*Motion Vectors*) associati ed è consentita l'ereditarietà tra essi, ovvero ogni PB può ereditare un MV appartenente ad un altro PB adiacente dal punto di vista spaziale o temporale.

Indipendentemente dalla scelta, le CB vengono ancora suddivise in TB (*Trasform Blocks*) costituenti TU (*Transform Units*). Su di esse viene effettuata la trasformata DCT (o una derivata dalla DST (*Discrete Sine Transform*)) ottenendo infine i coefficienti da codificare. Dopo la fase di quantizzazione, per la codifica entropica si usa una versione migliorata di CABAC (*Context Adaptive Binary Arithmetic Coding*). Essa condivide molto con quella usata in H.264 (Huffman infatti è stato via via abbandonato nelle codifiche video odierne), sebbene le prestazioni siano state migliorate. Per rendere più fedele possibile all'originale l'immagine in fase di ricostruzione diminuendo la quantità di artefatti dovuti alla blocchettizzazione vengono effettuati filtraggi chiamati DBLK (*Deblocking*, presente anche in H.264) e SAO (*Sample Adaptive Offset*).

Precedentemente è stato accennato come vantaggio di HEVC lo sfruttamento delle architetture multi-processore. La parallelizzazione necessaria è data in primo luogo dall'uso delle cosiddette *tiles*. Una tile è una regione di un'immagine codificabile e decodificabile in modo indipendente dalle altre. Un frame può essere diviso in tiles segmentandola in regioni rettangolari aventi tutte lo stesso numero di CTU. La mancanza di dipendenza tra esse fa in modo che possano essere processate in diversi thread. In alternativa è possibile utilizzare WPP (*Wavefront Parallel Processing*). In questo caso si fa riferimento alla suddivisione dei frame in slice, come già noto da H.264. Ogni slice viene divisa in righe di CTU ed ognuna di esse viene processata in modo indipendente.

Le principali novità di HEVC che sono state finora sinteticamente presentate danno un'idea del livello di complessità ma allo stesso tempo di flessibilità che questo codec è in grado di offrire. Nel capitolo successivo verrà effettuato un confronto sperimentale tra H.265 ed il predecessore H.264.

3

Analisi di H.264 e H.265

In questo capitolo si analizzeranno e si confronteranno le performance dei codec H.264 / AVC e H.265 / HEVC nella codifica e decodifica di sequenze video con risoluzione 4K e profondità di colore a 10 bit (HDR).

3.1 Scelte progettuali per l'analisi

3.1.1 Video di prova

L'analisi ha richiesto l'utilizzo di sequenze video standard allo scopo di rendere la prova facilmente riproducibile. A tal proposito è stato utilizzato il materiale riportato sul sito web [8]. Sono state considerate più di una sequenza video in modo da valutare le prestazioni dei due standard di codifica in modo accurato e in più situazioni. Di seguito le caratteristiche tecniche dettagliate dei filmati utilizzati per la prova.

1. `Netflix_TunnelFlag_4096x2160_60fps_10bit_420`

   Risoluzione: 4096x2160 pixel

   Frame rate: 60 frame per secondo

   Profondità di colore: 10 bit (High Dynamic Range, HDR)

   Spazio colore: YCbCr

   Chroma subsampling: 4:2:0

   Bitrate: 12740.2 Mbit/s

2. `Netflix_DrivingPOV_4096x2160_60fps_10bit_420`

   Risoluzione: 4096x2160 pixel

   Frame rate: 60 frame per secondo

   Profondità di colore: 10 bit (High Dynamic Range, HDR)

   Spazio colore: YCbCr

   Chroma subsampling: 4:2:0

   Bitrate: 12729.6 Mbit/s

3. `Netflix_SquareAndTimelapse_4096x2160_60fps_10bit_420`

   Risoluzione: 4096x2160 pixel

   Frame rate: 60 frame per secondo

   Profondità di colore: 10 bit (High Dynamic Range, HDR)

   Spazio colore: YCbCr

   Chroma subsampling: 4:2:0

   Bitrate: 12740.2 Mbit/s

4. `Netflix_ToddlerFountain_4096x2160_60fps_10bit_420`

   Risoluzione: 4096x2160 pixel

   Frame rate: 60 frame per secondo

   Profondità di colore: 10 bit (High Dynamic Range, HDR)

   Spazio colore: YCbCr

   Chroma subsampling: 4:2:0

   Bitrate: 12729.6 Mbit/s

I file video utilizzati sono in formato Y4M, la cui struttura verrà qui brevemente presentata.

I primi 10 byte contengono l'informazione riguardo il formato del file, ovvero la firma "YUV4MPEG2" seguita da uno spazio (con codifica ASCII esadecimale 0x20). Successivamente sono presenti parametri separati anch'essi da uno spazio:

- **Larghezza del frame:** W4096
- **Altezza del frame:** H2160
- **Frame rate:** F seguito dal numero di frame per secondo, espresso come una frazione nella forma "Numeratore:Denominatore", ovvero F60:1 per un frame rate di 60 Hz
- **Modalità di interlacciamento:** I seguita da una lettera che indica la modalità, in questo caso Ip dato che il video usa la modalità progressiva
- **Rapporto d'aspetto dei pixel:** A1:1 per pixel quadrati
- **Spazio colore:** C420p10, indica che il chroma subsampling è 4:2:0 e la profondità di colore è a 10 bit

Successivamente iniziano i byte che rappresentano i frame in formato YCbCr, ognuno preceduto da 5 byte con la firma "FRAME" seguita da uno spazio. La terminazione è data dalla configurazione esadecimale 0x0A.

Per approfondire la struttura dei file in formato Y4M nei casi più generali si rimanda a [5].

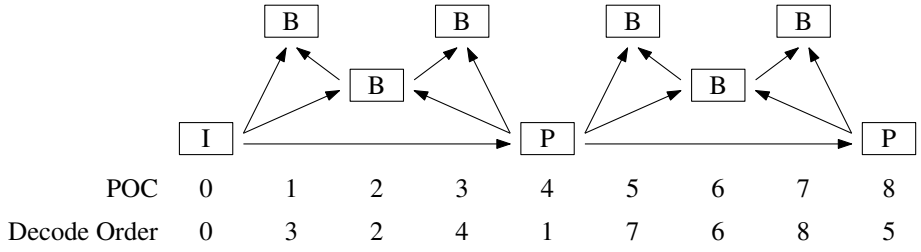

Figura 3.1.: Struttura del GOP utilizzato

Per rendere i file video adatti all'uso con gli encoder è stata necessaria una conversione dal formato Y4M al formato YUV dal momento che il codificatore accetta in input la sequenza di frame byte per byte senza alcun header. Per far ciò è stato utilizzato lo script MATLAB presente in Appendice 1.

I frame ottenuti da questo processo hanno quindi una dimensione (non compressa) di 4096 x 2160 x 2 byte x 3 / 2 = 26542080 byte. I 2 byte derivano dal fatto che ogni pixel necessita di 10 bit invece dei canonici 8 per la rappresentazione. Il fattore 3/2 deriva invece dallo schema di decimazione in uso, ovvero 4:2:0. La componente di luminanza avrà dimensione effettiva di 4096 x 2160, mentre le due componenti di crominanza avranno una dimensione dimezzata sia in verticale che in orizzontale, portando l'area ad essere 1/4. Da qui 1 + 1/4 + 1/4 = 3/2.

3.1.2 Encoder e relative configurazioni

Gli encoder utilizzati per la prova sono quelli di riferimento, ovvero HM (HEVC Test Model) 16.14 e JM 19.0, liberamente scaricabili dalle pagine [6] e [7] del sito web del Fraunhofer Institute. Essi sono stati compilati con il compilatore Microsoft Visual C++ per essere eseguiti su architetture a 64 bit, in modo da sfruttare al meglio le peculiarità del processore in uso e poter utilizzare una quantità maggiore di memoria di sistema (RAM) se necessario.

Entrambi gli encoder sono stati impostati per utilizzare come Group of Pictures (GOP) quello presentato nella relativa documentazione, il cui schema è riportato in Figura 3.1.

Per quanto riguarda il fattore di quantizzazione, due approcci diversi sono stati seguiti per i due codificatori:

- H.264 è stato configurato per mantenere costante il fattore di quantizzazione in base al tipo di frame da codificare. I valori impostati sono stati 32 per i frame di tipo I, 33 per quelli di tipo P e 34 per quelli di tipo B.

- H.265 invece ha optato per un'ottimizzazione dei fattori di quantizzazione tramite RDO (Rate-Distortion Optimization), usando 32 come base.

La differenza qui riportata non è tale da creare particolari problemi nel confronto tra i due video codificati, in quanto si è osservato che l'encoder HEVC ha scelto autonoma-

mente valori del fattore di quantizzazione compresi tra 32 e 35, quindi molto vicini se non uguali a quelli utilizzati da AVC.

Gli algoritmi utilizzati per la predizione sono stati TZ search per H.265 e EPZS (Enhanced Predictive Zonal Search) per H.264. Essi non richiedono l'analisi di ogni blocco del fotogramma considerato ma solo di quelli vicini a quello di cui si esegue la predizione.

Per ogni video oggetto dell'analisi sono stati codificati i primi 10 frame, per una durata quindi di 1/6 di secondo. Questa quantità è stata considerata un buon compromesso in quanto consente di determinare con precisione sufficiente tutti i parametri che verranno successivamente indicati senza richiedere tempi di elaborazione eccessivi.

3.1.3 Piattaforma hardware e software

La configurazione hardware su cui sono stati effettuati i test è basata su un processore a 4 core (8 thread) Intel Core i7 3770 con frequenza base di funzionamento di 3.4 GHz. In condizioni di carico elevato può raggiungere i 3.9 GHz. Il sistema operativo utilizzato è Windows 10 Pro 1703 a 64 bit. La piattaforma utilizzata per la prova non è dotata di supporto hardware ad H.265, mentre è presente quello ad H.264. Quest'ultimo è stato quindi volutamente disabilitato, in modo da ottenere risultati confrontabili.

3.2 Risultati dell'analisi

I parametri considerati per la comparativa sono:

- Qualità oggettiva
- Bitrate richiesto (e quindi entità della compressione)
- Prestazioni in termini di tempo necessario per la codifica

3.2.1 Qualità oggettiva

Per determinare la qualità dal punto di vista oggettivo i due indicatori di riferimento utilizzati sono stati PSNR e SSIM.

PSNR (Peak Signal to Noise Ratio) è definito come il rapporto tra la massima potenza del segnale considerato e la potenza del rumore. Il suo valore si basa sullo scarto quadratico medio (MSE). Dato X il fotogramma del video originale ed Y quello del fotogramma codificato, possiamo scrivere:

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [X(i,j) - Y(i,j)]^2$$

$$PSNR_{dB} = 10 \log_{10} \frac{1023^2}{MSE}$$

dove m ed n sono rispettivamente la larghezza e l'altezza di un singolo fotogramma.

Nella formula del PSNR riportata è facile intuire che il numero 1023 al numeratore è strettamente legato ai 10 bit di Color Bit Depth.

Nel caso in esame il PSNR in dB viene calcolato direttamente dall'encoder, sia come valore per ogni componente, sia come media tra essi.

SSIM (Structural SIMilarity) è un indice basato sulla qualità effettivamente percepita dall'occhio umano. A differenza del PSNR infatti tiene conto delle peculiarità dell'HVS risultando così un indice più significativo. Esistono infatti situazioni in cui la qualità dell'immagine dopo la compressione è alta pur avendo un PSNR molto basso o viceversa. Può assumere un valore compreso tra 0 e 1 (dove 1 indica che le immagini da confrontare sono identiche) ed è ottenuto usando la formula riportata di seguito:

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}$$

SSIM è definito solo per immagini statiche. Perciò quello che verrà riportato nei risultati delle prove sarà la media tra i valori di SSIM di ogni fotogramma di cui è costituito il video analizzato.

I risultati sperimentali divisi per componente e medi sono contenuti nelle tabelle 3.1, 3.2, 3.3 e 3.4.

Video 1	**PSNR-Y**	**PSNR-Cb**	**PSNR-Cr**	**PSNR**	**SSIM**
H.265 / HEVC	38.241	43.123	45.078	39.4089	0.8803
H.264 / AVC	36.972	42.508	44.677	38.195	0.8678

Tabella 3.1.: Qualità oggettiva relativa al Video 1

Video 2	**PSNR-Y**	**PSNR-Cb**	**PSNR-Cr**	**PSNR**	**SSIM**
H.265 / HEVC	37.334	43.193	39.730	38.260	0.9246
H.264 / AVC	36.262	42.974	39.593	37.319	0.9168

Tabella 3.2.: Qualità oggettiva relativa al Video 2

Video 3	**PSNR-Y**	**PSNR-Cb**	**PSNR-Cr**	**PSNR**	**SSIM**
H.265 / HEVC	39.828	45.191	44.009	40.894	0.9320
H.264 / AVC	37.525	44.094	42.853	38.726	0.9144

Tabella 3.3.: Qualità oggettiva relativa al Video 3

Video 4	**PSNR-Y**	**PSNR-Cb**	**PSNR-Cr**	**PSNR**	**SSIM**
H.265 / HEVC	34.989	43.673	39.536	35.987	0.8845
H.264 / AVC	33.034	43.852	39.571	34.067	0.8584

Tabella 3.4.: Qualità oggettiva relativa al Video 4

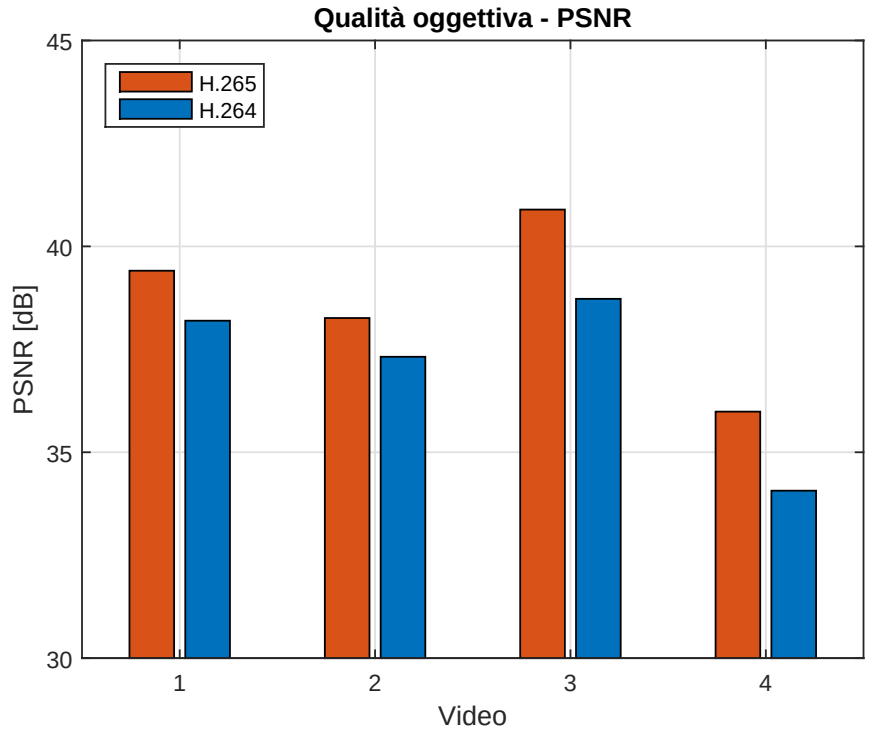

Figura 3.2.: Grafico per il confronto del PSNR

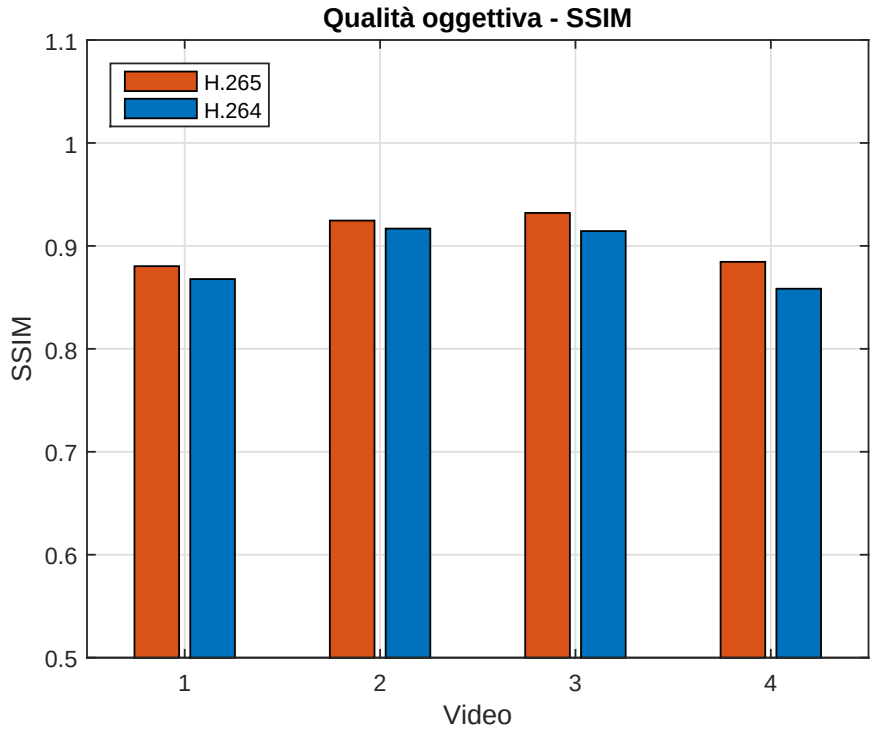

Figura 3.3.: Grafico per il confronto di SSIM

I due indici considerati sono in accordo tra loro e rilevano come HEVC sia in grado di produrre un file video codificato di maggior qualità oggettiva. Sebbene la differenza

possa sembrare solo marginale, non bisogna dimenticare che il PSNR è espresso in dB e quindi utilizza la scala logaritmica. Una variazione di 1 dB tra H.265 e H.264 comporta che in scala lineare il rapporto segnale-rumore di HEVC sia circa 1.26 volte quello di AVC, una differenza quindi rilevante.

Le precedenti considerazioni valgono per tutti i video di prova considerati. Si può osservare che ad un valore minore di PSNR non corrisponde necessariamente un valore minore di SSIM. I primi due video sono un chiaro esempio. Inoltre il secondo video presenta una variazione più contenuta del valore di SSIM tra la codifica H.265 e H.264 quando messo a confronto con il primo, il terzo ed il quarto video. Ciò comporta una differenza di qualità visiva ancor meno rilevante.

3.2.2 Bitrate e dimensione

Video 1	**Bitrate**	**Dimensione**	**Variazione**
H.265 / HEVC	16624.56 kbit/s	346396 byte	-57%
H.264 / AVC	38748.00 kbit/s	807250 byte	

Tabella 3.5.: Bitrate e dimensione relativi al Video 1

Video 2	**Bitrate**	**Dimensione**	**Variazione**
H.265 / HEVC	13109.18 kbit/s	273159 byte	-38.29%
H.264 / AVC	21246.10 kbit/s	442627 byte	

Tabella 3.6.: Bitrate e dimensione relativi al Video 2

Video 3	**Bitrate**	**Dimensione**	**Variazione**
H.265 / HEVC	19869.89 kbit/s	414007 byte	-28.17%
H.264 / AVC	27664.37 kbit/s	576341 byte	

Tabella 3.7.: Bitrate e dimensione relativi al Video 3

Video 4	**Bitrate**	**Dimensione**	**Variazione**
H.265 / HEVC	95724.00 kbit/s	1994301 byte	-11.44%
H.264 / AVC	108092.30 kbit/s	2251923 byte	

Tabella 3.8.: Bitrate e dimensione relativi al Video 4

Un buon algoritmo di compressione video permette di raggiungere una qualità dell'immagine predeterminata con un bitrate minore possibile. Per ogni video oggetto dell'analisi, HEVC dimostra la sua superiorità, richiedendo un bitrate e di conseguenza un'occupazione in memoria minori rispetto ad AVC. La differenza è netta nei primi tre casi, molto minore nel quarto. I valori numerici sono riportati nelle tabelle 3.5, 3.6, 3.7 e 3.8. Il campo "Variazione" indica di quanto si discosta il valore di bitrate relativo a HEVC nei confronti di AVC. Ciò corrisponde anche alla differenza di dimensione dato che le due grandezze sono strettamente legate a parità di numero di frame considerati.

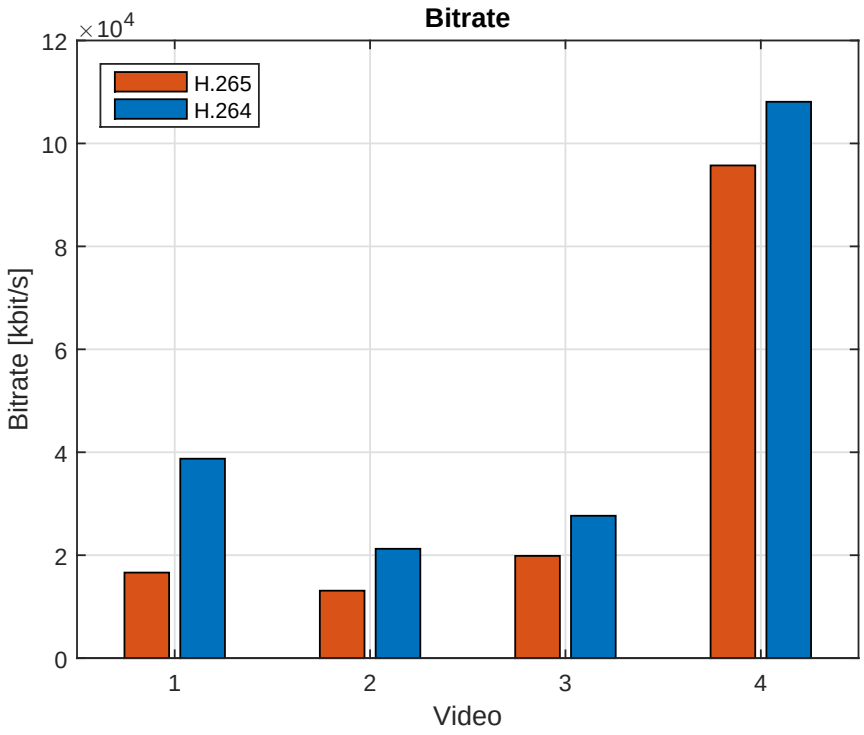

Figura 3.4.: Grafico per il confronto del bitrate

È evidente come il diverso tipo di scena rappresentata influisca pesantemente sulla capacità di compressione. La prima sequenza video, analogamente alla seconda, riporta una scena stradale in cui il movimento di tutti gli oggetti è facilmente prevedibile. Nella terza invece ogni persona facente parte della folla ha un moto non determinabile a priori con certezza. Infine il caso peggiore è rappresentato dalla quarta sequenza video, la quale contiene numerosi elementi in primo piano (gli spruzzi di una fontana) le cui caratteristiche geometriche sono in continua variazione.

Si può quindi affermare che HEVC è effettivamente in grado di raddoppiare la capacità di compressione rispetto al predecessore, sebbene ciò non sia possibile in ogni situazione.

Considerando in modo particolare le prime due sequenze video, risultati ancora migliori si sarebbero potuti raggiungere con un GOP formato da un numero maggiore di immagini rispetto alle 4 utilizzate. Ciò avrebbe portato ad un miglior sfruttamento delle nuove potenzialità offerte da HEVC in termini di tecniche di predizione, ma d'altra parte avrebbe probabilmente causato una perdita di definizione dell'immagine con un conseguente abbassamento dei valori di PSNR e SSIM.

3.2.3 Prestazioni temporali

La complessità computazionale gioca un ruolo fondamentale nella compressione video. Infatti, oltre ad una buona qualità visiva e ad una minima quantità di spazio di archiviazione richiesta, il tempo necessario per la codifica della sequenza è non meno importante. In questo ambito HEVC non eccelle, richiedendo un tempo per la codifica molto maggiore rispetto ad H.264. Si riscontra infatti un aumento che va da un minimo del 295% ad un massimo del 488%. I dati completi sono presentati nelle tabelle 3.9, 3.10, 3.11 e 3.12, dove il campo "Variazione" indica di quanto si discosta il tempo necessario relativo a HEVC nei confronti di AVC.

Questo accade nonostante l'encoder H.265 sia in grado di sfruttare maggiormente le potenzialità del processore multi-core in uso, come infatti si è potuto constatare durante il test.

Risultati nettamente migliori si possono certamente ottenere delegando parte del lavoro di codifica ad una GPU con supporto a HEVC a livello di hardware, come già accennato nel paragrafo 2.5.4.

Tuttavia è bene osservare che l'utilizzo di memoria RAM da parte di H.264, sebbene non sia qui riportato in quanto molto variabile durante il processo di codifica, è stato molto maggiore rispetto ad H.265, arrivando anche a valori 10 volte superiori.

Video 1	**Tempo**	**Variazione**
H.265 / HEVC	1071.424 s	+384.25%
H.264 / AVC	239.024 s	

Tabella 3.9.: Prestazioni temporali relative al Video 1

Video 2	**Tempo**	**Variazione**
H.265 / HEVC	837.017 s	+295.47%
H.264 / AVC	211.653 s	

Tabella 3.10.: Prestazioni temporali relative al Video 2

Video 3	**Tempo**	**Variazione**
H.265 / HEVC	949.530 s	+391.29%
H.264 / AVC	226.464 s	

Tabella 3.11.: Prestazioni temporali relative al Video 3

Video 4	**Tempo**	**Variazione**
H.265 / HEVC	1716.863 s	+488.29%
H.264 / AVC	291.835 s	

Tabella 3.12.: Prestazioni temporali relative al Video 4

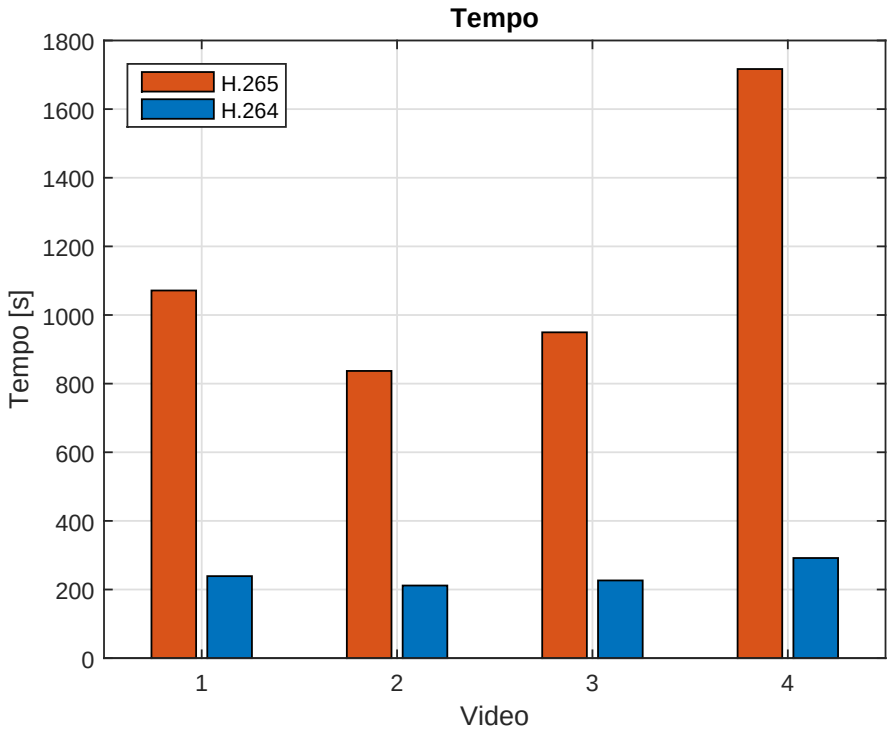

Figura 3.5.: Grafico per il confronto del tempo di codifica

3.2.4 Rapporto PSNR-Bitrate

Oltre ai test riportati nelle sezioni precedenti, i quali consideravano tutti i parametri di interesse mantenendo un valore di QP (Quantization Parameter) pari a circa 32, sono state effettuate ulteriori prove assegnando a QP i valori 10, 20, 30, 40 e 50 (esso infatti può variare da un minimo di 0 ad un massimo di 51, dove tendendo verso 0 si ha una quantizzazione molto fine mentre verso 51 una quantizzazione sempre più grossolana).

Sono stati quindi prodotti quattro grafici (Figure 3.6, 3.7, 3.8 e 3.9), uno per ogni sequenza video in analisi, che indicano la variazione del PSNR in funzione del bitrate.

Analizzandoli si nota che al diminuire del parametro QP (e di conseguenza all'aumentare del bitrate richiesto) la differenza tra i due standard di codifica si fa sempre maggiore. Particolarmente rilevante è il risultato ottenuto con il secondo video, dove con QP = 10 si ha un PSNR simile ma un bitrate più che dimezzato a favore di HEVC. Anche il quarto video, che in precedenza si è dimostrato problematico per H.265, mostra che con valori di QP pari a 10 e 20 il nuovo standard consente di ottenere un PSNR simile o addirittura superiore con un bitrate inferiore di poco più del 20%.

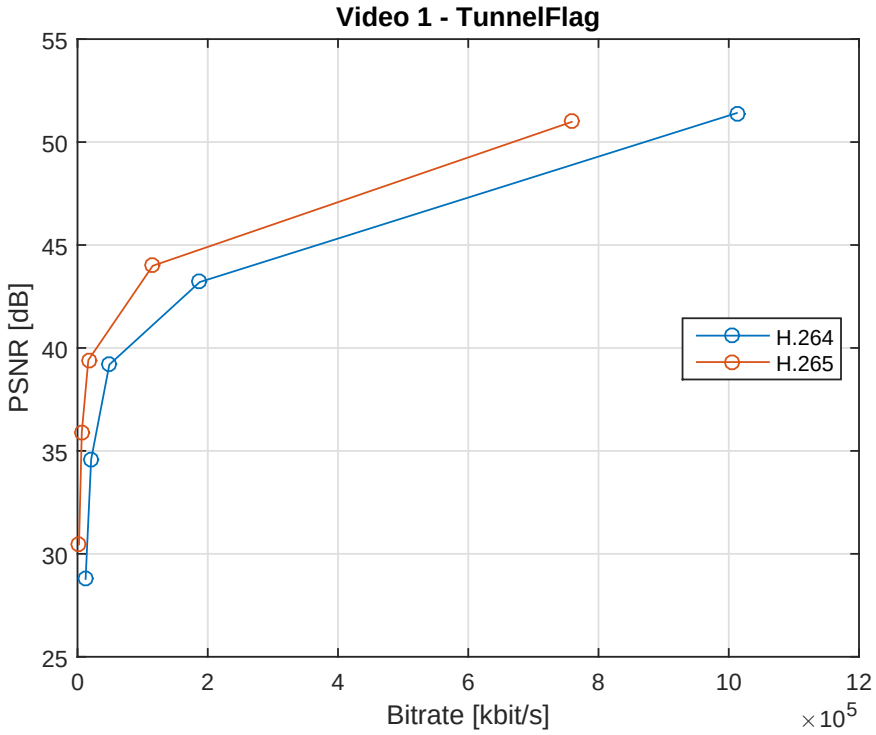

Figura 3.6.: Grafico PSNR/Bitrate relativo al video 1

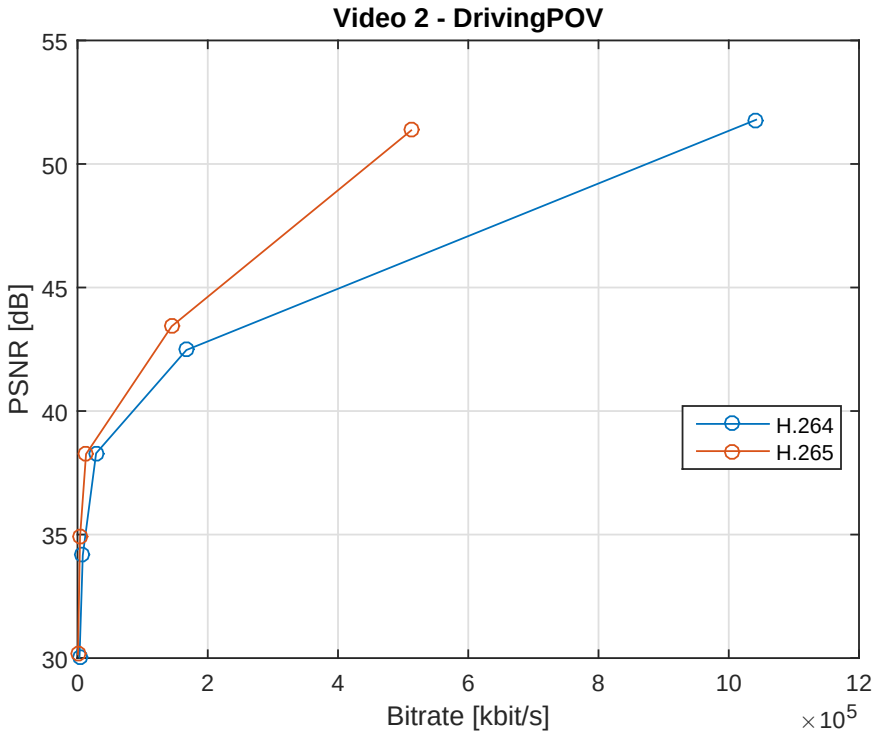

Figura 3.7.: Grafico PSNR/Bitrate relativo al video 2

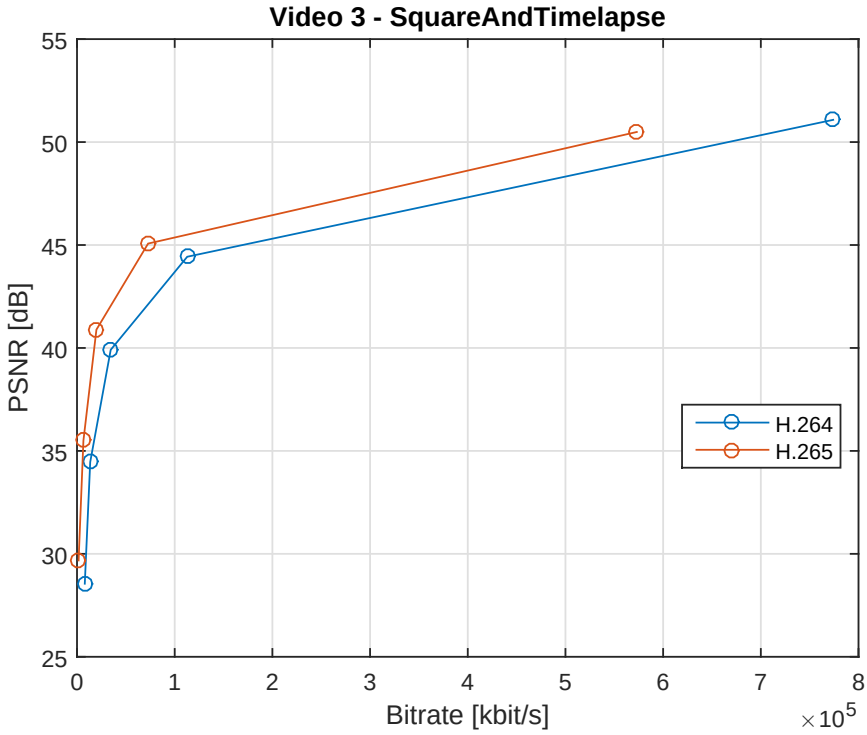

Figura 3.8.: Grafico PSNR/Bitrate relativo al video 3

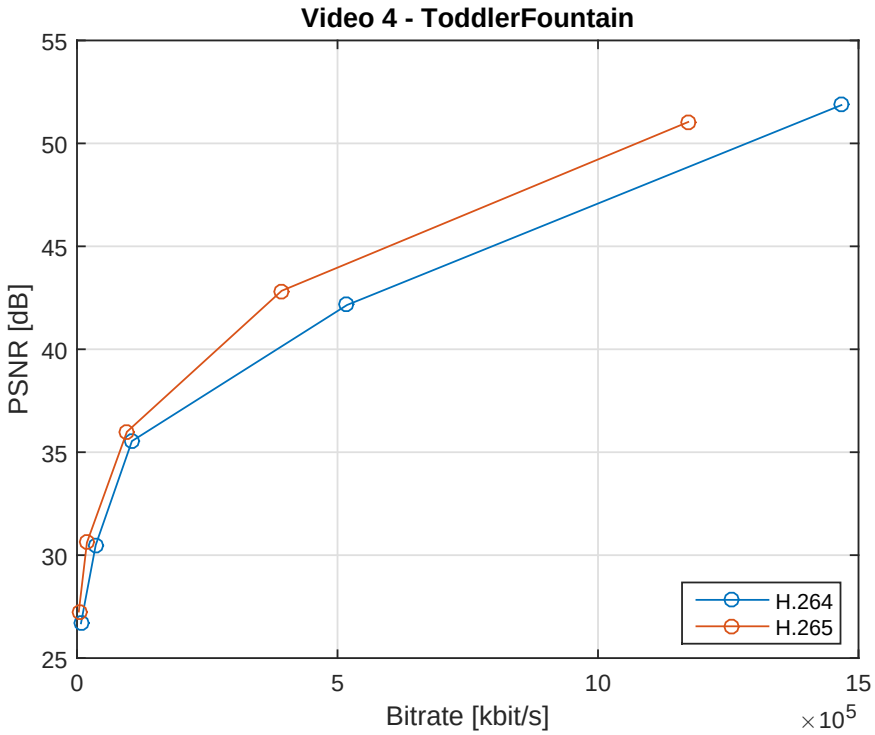

Figura 3.9.: Grafico PSNR/Bitrate relativo al video 4

4

Conclusioni

I risultati sperimentali confermano la bontà sotto numerosi aspetti della codifica HEVC. La maggior parte dei valori riguardanti H.265 è infatti migliore di quelli ottenuti da H.264. Sebbene dal punto di vista della qualità oggettiva il cambiamento non sia molto marcato anche se non trascurabile, notevoli sono invece i progressi in termini di bitrate richiesto e di dimensione del file di output. Il tempo di elaborazione necessario fa eccezione, ma come si è osservato ciò è facilmente risolvibile utilizzando una piattaforma in grado di eseguire la codifica via hardware. Implementazioni diverse da quella di riferimento inoltre possono portare a risultati di livello superiore.

Bisogna ricordare che questo risultato è frutto di una quantità sempre maggiore di migliorie che vengono apportate alla procedura di base illustrata nel Capitolo 1. Essa, pur essendo nata alla fine degli anni '80 del secolo scorso, dimostra tutt'ora la sua validità essendo infatti ispirazione per ogni nuovo standard di codifica video.

HEVC sarà certamente lo standard di riferimento per gli anni a venire, essendo le trasmissioni 4K sempre più diffuse. Con l'aggiunta del supporto all'HDR su cui si sta puntando molto anche in ambito consumer la transizione da H.264 a H.265 potrà inoltre subire un'accelerazione.

Non essendo l'adozione di HEVC ancora molto estesa e non avendo trovato limiti nelle applicazioni odierne, il suo successore non è ancora oggetto di discussione. Tuttavia ad esso si contrappongono altri codec quali VP9 e AV1 (AOMedia Video 1), i quali si differenziano da H.265 per essere *royalty-free*.

A

Appendice 1

Codice Matlab per la conversione di un file video dal formato Y4M a YUV

```
function [] = convY4MYUV(inName, outName, width, height, numFrames)

inF = fopen(inName,'rb');    %Apertura file di input (modalita' lettura di byte)
outF = fopen(outName,'wb'); %Apertura file di output (modalita' scrittura di byte)

t = fseek(inF,50,'bof');     %Imposta la posizione del cursore del file di input, saltando 50 byte a partire dall'inizio del file
for i = 1:numFrames          %Esegue la stessa operazione per ogni frame
    frameBytes = fread(inF,[(width*height*3) 1],'uint8');     % Legge dal file in input i byte del frame i-esimo
    count = fwrite(outF,frameBytes,'uint8');                  % Scrive nel file di output i byte letti dal frame
    t = fseek(inF,6,'cof'); %Sposta il cursore di 6 byte per saltare la firma del frame successivo
end;

fclose(inF);
fclose(outF);
```

Bibliografia

[1] Iain E. Richardson, "The H.264 Advanced Video Compression Standard", 2 ed., *John Wiley Sons*, 2010

[2] Vivienne Sze, Madhukar Budagavi, Gary J. Sullivan, "High Efficiency Video Coding (HEVC): Algorithms and Architectures", *Springer International Publishing*, 2014

[3] Gary J. Sullivan, Jens-Rainer Ohm, Woo-Jin Han, Thomas Wiegand, "Overview of the High Efficiency Video Coding (HEVC) Standard", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1649-1668, Dicembre 2012

[4] Keith Jack, "Video Demystified: A Handbook for the Digital Engineer", 4 ed., *Elsevier*, 2005

[5] "YUV4MPEG2", Giugno 2017, https://wiki.multimedia.cx/index.php/YUV4MPEG2

[6] "ITU-T Recommendation H.265.2: Reference software for ITU-T H.265 high efficiency video coding", Novembre 2016, https://hevc.hhi.fraunhofer.de/

[7] "H.264/AVC reference software", Giugno 2015, http://iphome.hhi.de/suehring/tml/

[8] "Xiph.org Video Test Media", https://media.xiph.org/video/derf/