

Codifica Video

- Successione di foto (frame)
- C'è molta ridondanza tra un frame e il successivo

Quanto occupa un video

Considero un segnale televisivo analogico standard (PAL) che transita attraverso un sistema di conversione analogico-digitale (componatore + quantizzatore a 65536 livelli).

$$B = 6\text{MHz} \rightarrow f_c = 12\text{MHz}$$

componamento

$$\# \text{bit} = \log_2(65536) = 16 \Rightarrow 2^{16} = 65536$$

$$\text{bit-rate}^{(R)} = 16 \cdot f_c = 192\text{Mbits}$$

In un minuto sono trasmessi $R \cdot 60\text{s} = 1.14\text{Gbyte}$

Si usano in comune codifiche lossy e loss-less

Standard video

- MPEG 1: sistemi a banda larga, video cd (1.5 Mbit/s)
- MPEG 2: sistemi a banda larga, DVD e digitale terrestre (10 Mbit/s) DVB
- MPEG 4: sistemi a banda stretta, telefoni UMTS (1998)

La codifica MPEG 2 originariamente pensata per arrivare ad un segnale di 10 Mbit/s in realtà è strutturata per ^{potere} superare anche e raggiungere i 40 Mbit/s che si ottengono dallo standard MPEG-3 (obsoleto)

- MPEG 7
 - MPEG 21
- Non per codifica video, ma per ricerca dettagli video. Già MPEG 2/4 non ha le caratteristiche necessarie a far fronte a tutte le richieste del mercato

MPEG è standard aperto

Le immagini all'interno di un video sono codificate in tre modi e secondo due forme, ossia:

→ Intra-frame (frame I)^(chiave): frame chiave, codificati J-P-E-G. Sono codificati in maniera indipendente rispetto a tutti gli altri frame (switch scene film)

Supponendo di switchare da un frame I al suo successivo si nota che fra i due frame ci sono altre foto (frame) con molta ridondanza, per se ho un bit-rate di 24 allora la foto cambia ogni $\frac{1}{24}$ di secondo ossia rispetto alla precedente ha quasi lo stesso contenuto informativo. Quindi rispetto al frame I precedente ho 90-95% di info uguali → trasmetto cioè la differenza rispetto al precedente (rimozione ridondanza temporale). Poiché la trasmissione dello differenza rispetto al precedente non è efficiente devo introdurre altri 2 frame.

→ Inter frame (frame P)_(OP): codificato con riferimento alle immagini precedenti ossia è ottenuto per differenza dal frame I di partenza

→ Frame B: codificato con riferimento al frame successivo e al precedente ossia è codificato a doppia predizione una nel passato e una nel presente. → Dipende da I e P. (doppia diff.)

Le predizione doppia si può fare su un numero limitato di frame e non su tutto il video. (Altrimenti aumento troppo l'error)

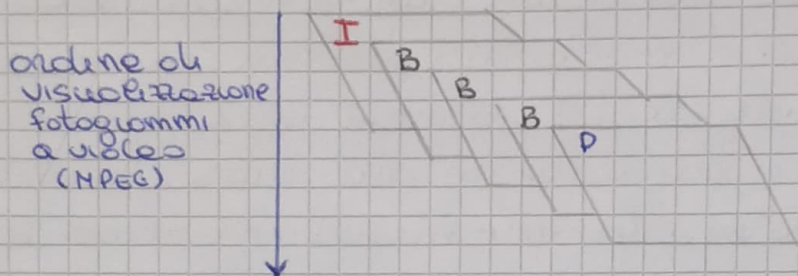
⇒ Perciò il filmato viene partizionato in gruppi di immagini (GOP) che iniziano e finiscono con un frame chiave (frame I) ⇒ Le scene.

Il fatto che il GOP sia chiuso da un frame I serve anche per permettere la sincronizzazione audio.

Il GOP è di solito costituito da 10/15 immagini e al

suo interno troviamo i frame ottenuti per differenze.

Limite a 15 foto perché i frame P (eccetto il primo che si riferisce al frame I) vengono codificati sui precedenti frame P. Quindi se aumento troppo il numero di frame P dedotti da altri frame P l'errore di risoluzione aumenta \Rightarrow bassa qualità.



Le foto in fase di trasferimento sono trasferite diversamente secondo l'ordine: **IPBBBP...**

Il decoder dunque riceve I per l'informazione per ricostruire P (che ottiene sommando I con i dati di P) e infine riceve l'informazione per ricostruire B. Alla fine il decoder riordina le foto in ordine MPEG e li manda a video (50-100 ms).

- Per trasmissione in banda stretta (video call) 80/100 ms. per avere il video e schema sono eccessivi perciò non vengono trasmessi i frame B.

Predizione

- fornisco un vettore di movimento (motion vector) che mi spiega come gli elementi si sono mossi, (coordinate movimento)
- L'immagine viene suddivisa in macroblocchi 16×16 e in ogni macroblocco si ricerca il vettore movimento
- Come capire dove l'oggetto si è spostato e la parte aperta dello storciador. Valore positivo significa movimento da verso il basso e negativo per spostamento in alto o a sinistra.

Oltre al vettore di movimento viene inviata l'informazione relativa alla **Moto-Compensazione** che informa il ricevitore delle eventuali rotazioni dell'immagine. **Retroazione ad anello chiuso**: schema usato da tutti i codificatori in ambito multimediale:

→ il trasmettitore, che conosce sia il file originale sia i dati trasmessi al ricevitore per ricostruire la foto, ricostruisce il file come è in grado di ricostruirlo il ricevitore e analizza l'errore della ricostruzione;

=> esegue una differenza tra il frame originale e il frame ricostruito in base alle informazioni predette.

In questo caso la differenza tra i frame rappresenta l'informazione di moto-compensazione che poi sarà trasmessa sul canale

- Sul frame chiave si applica la codifica JPEG quindi si suddivide in blocchi di 8×8 pixels
- Con la codifica J-Peg il frame I viene "spostato" dallo spazio di colore RGB a YUV così da mantenere il 90% delle informazioni utili sulle matrice di luminanza e di conseguenza poter sottocomprimere le 2 matrici di cromaticità.

Il sottocomprimimento di fatto rende impossibile recuperare il vettore spostamento nella cromaticità e di conseguenza verrà cercato sulle luminanze

La foto codificata JPEG dovrebbe avere perciò 4 matrici di luminanza (essendo suddivise in blocchi 16×16) e 4 di cromaticità. => Poiché la cromaticità è poco ricca di informazioni le **Sottocomprimono** e trasmettono solo 2 cromaticità (1 volta in UV e una volta V)

Sottocampionamento

$4:2:2$

- rapporto campionamento che eseguo:
 - 1° numero: sulle luminanze
 - 2° e 3° numero: sulle componenti di cromaticità

Uso cioè n. bit diversi per le 3 matrici:

→ il primo è il n. massimo, gli altri le sottocampionamenti orizzontalmente di un fattore 2 le componenti di cromaticità (secondo numero). Il terzo indica il sottocampionamento verticale (sulle colonne) delle 2 cromaticità. Se è pari a zero indica il sottocampionamento $2:1$

$4:2:0$: usato in JPEG, JFIF, H.261 e MPEG-1
ossia trasmette sempre la luminanza e alternandole trasmette una fra le 2 cromaticità

$4:2:2$: usato per i video MPEG-2

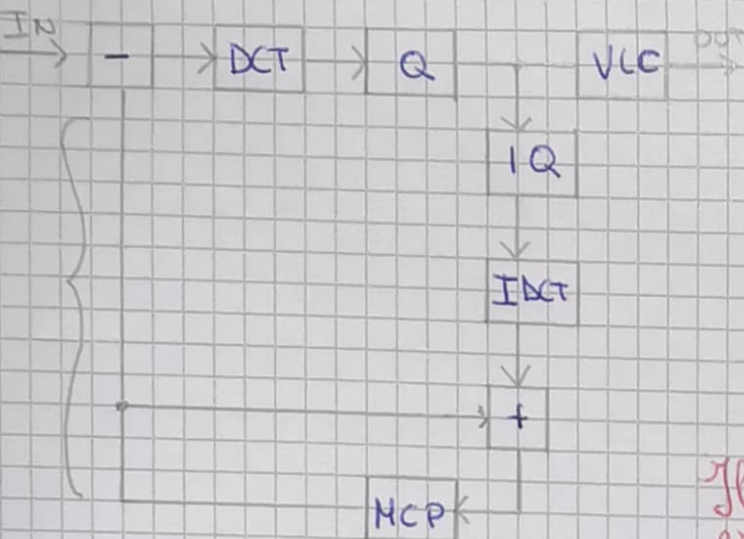
$4:0:0$: immagine bianco-nero

DCT

Dopo aver rimosso la ridondanza temporale si rimuove quella spaziale usando la dct di algoritmo jpeg

$$F(u,v) = \frac{C_u}{2} \frac{C_v}{2} \sum_{y=0}^7 \sum_{x=0}^7 f(x,y) \cos \left[\frac{(2x+1)u\pi}{16} \right] \cos \left[\frac{(2y+1)v\pi}{16} \right]$$

Codificatore MPEG-2



(D) DCT: trasformazione coseno discreta (Inverse)

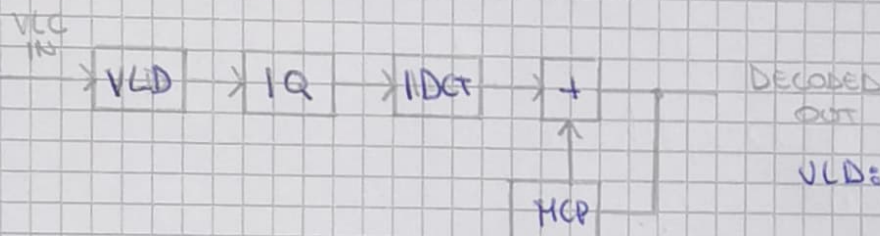
(I) Q: quantizzazione (Inverse)

VLC: codifica a lunghezza variabile (HUFFMAN)

MCP: Predizione moto compensativa

Il trasmettitore continua ad eseguire MCP finché l'errore in trasmissione non è sotto minimo

De-Codificatore MPEG-2



VLD: decodifica a lunghezza variabile

MPEG-4

→ banda stretta

→ Il frame si considera costituito da oggetti diversi che si fondono per formare la scena.

Quindi la codifica viene in base all'oggetto, e all'importanza dello stesso nella scena (+ primo piano)
- secondo piano

→ Vedo l'oggetto come e quanto si sposta quindi non ho più bisogno di dividere l'immagine in macro-blocchi ⇒ Auro' piani diversi non più immagini
ossia ho il piano indipendente I VOP e gli altri P e B-VOP

→ Posso manipolare l'immagine in 4 modi:

- Codifica: distribuzione bit rate x ogni oggetto
- Multiplozione: ottenzione scene
- De-Multiplozione: ricevo solo una parte della scena
- De-codifica: modifica composizione oggetti