UNIVERSITÀ
DEGLI STUDI
FIRENZE

Sviluppo di sistemi per compressione video semantica

Tintori Matteo

Relatore: Bertini Marco

Introduzione

Di cosa si parla

- Codec semantico per video di calcio HD
- Compressione basata su H.265
- Riconoscimento di zone di importanza maggiore tramite reti neurali
- Risparmio memoria e bitrate video codificato

Strumenti

Cosa abbiamo usato

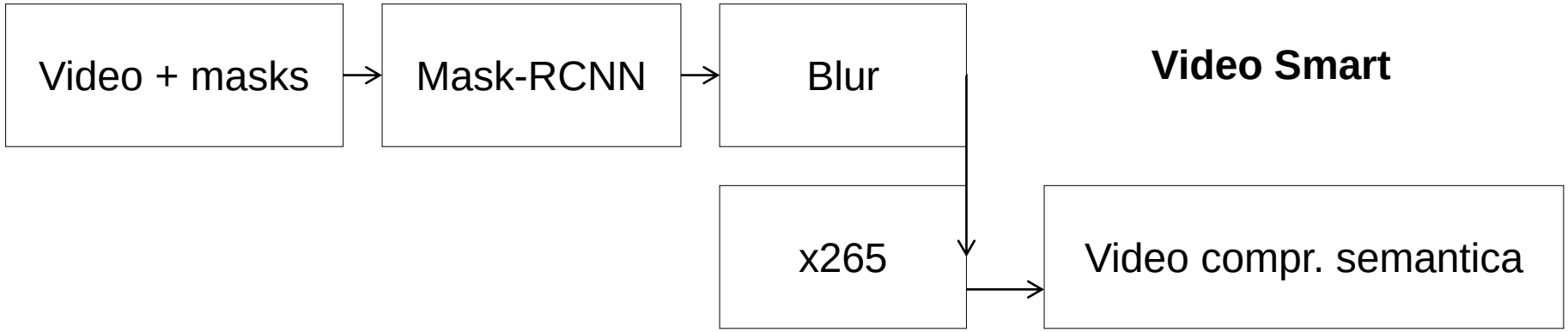
Una prima distinzione tra ciò di cui abbiamo fatto uso si può suddividere più dettagliatamente in:

- strumenti software
- strumenti hardware.

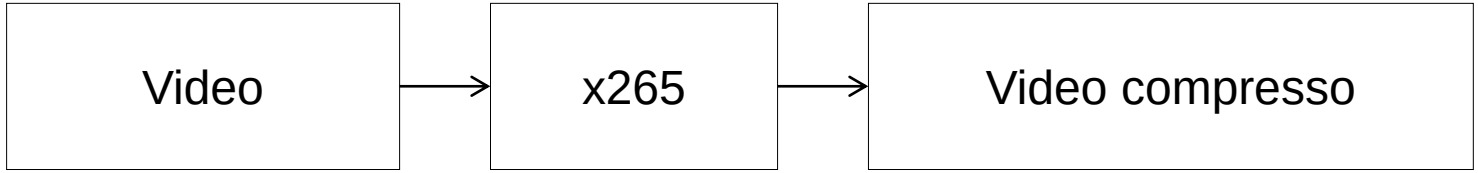
- Strumenti hardware: 2 GPU Nvidia Titan X
- Strumenti software:
- Codec usato: hevc_nvenc (standard H.265)
- Rete neurale Mask-RCNN: implementazione tensorflow (1.15.x) + Keras
- Libreria FFMPEG per codifica video

Sistema

Descrizione del sistema

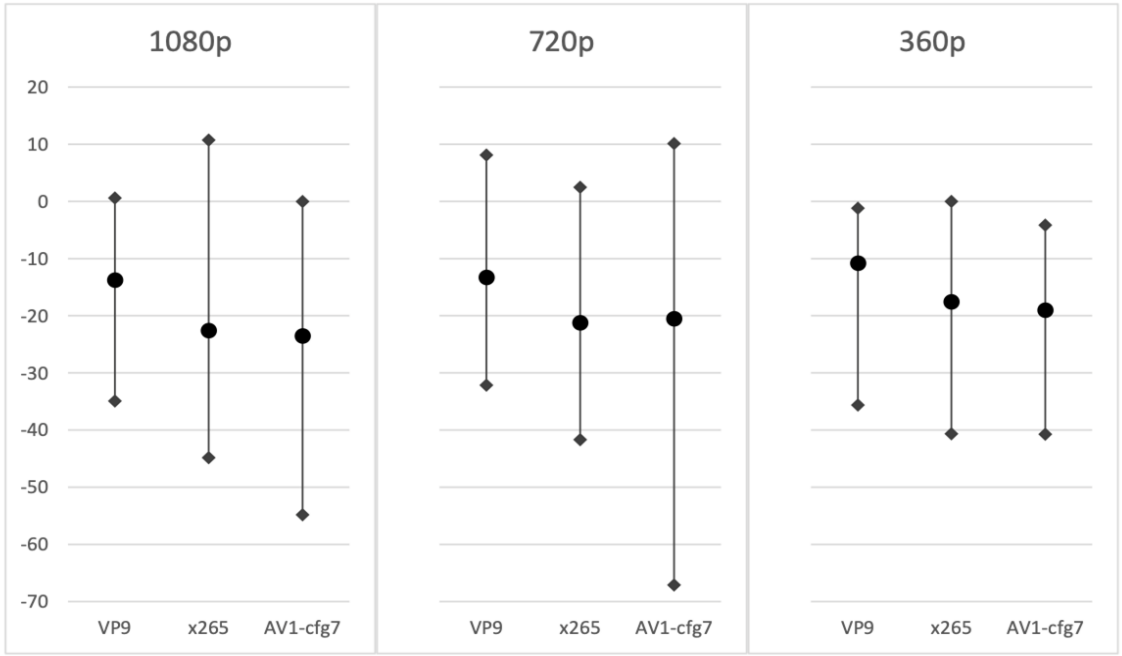


Video Standard



Codec

Base iniziale



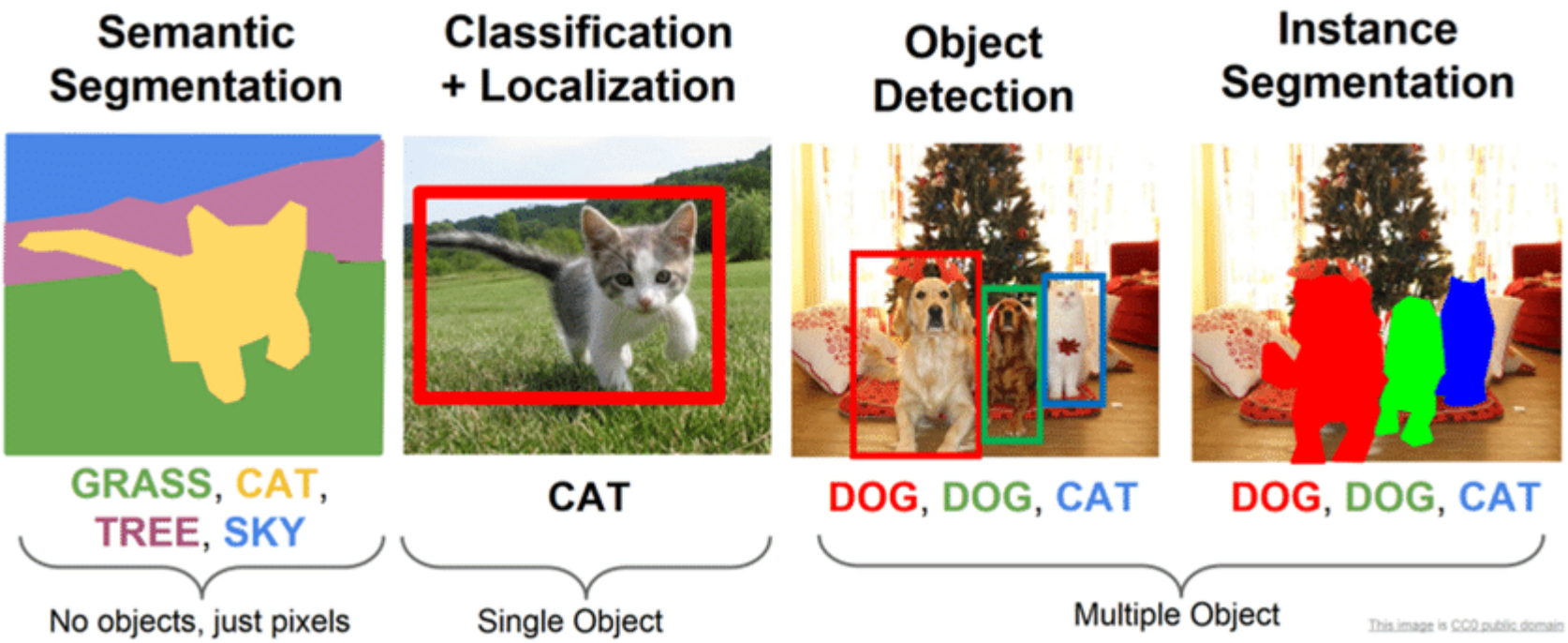
Asse x: codec
Asse y: risparmio in bits
(positivo se i numeri seguono il segno -)

- Paragone tra encoder rispetto alla qualità del video
- Vittoria di x265 in base al risparmio di bits

Mask-RCNN

Rete neurale adottata

- Mask-RCNN = Faster RCNN + FCN
- Preaddestrata per circa 60 classi
- Training per classi palla e giocatori
- Instance segmentation: le maschere vengono tracciate su ogni singolo oggetto con la relativa etichetta nella forma in cui l'oggetto si presenta.



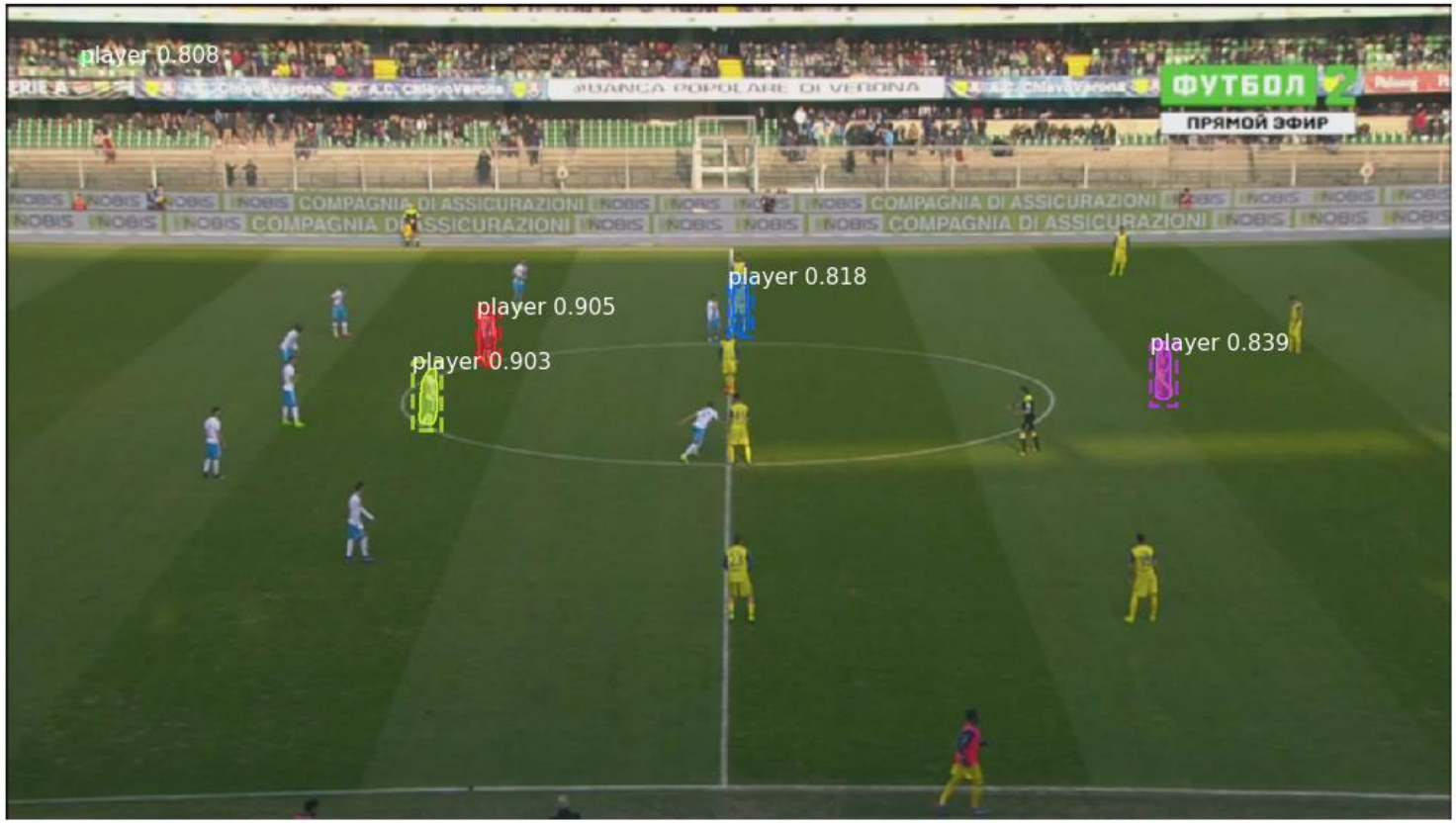
Labeling

Etichettatura immagini per l'allenamento

- Software di labeling LABELME per finetuning
- Produzione annotazioni intero dataset utilizzando tracciamento dei contorni.

Funzionalità

Previste e fornite

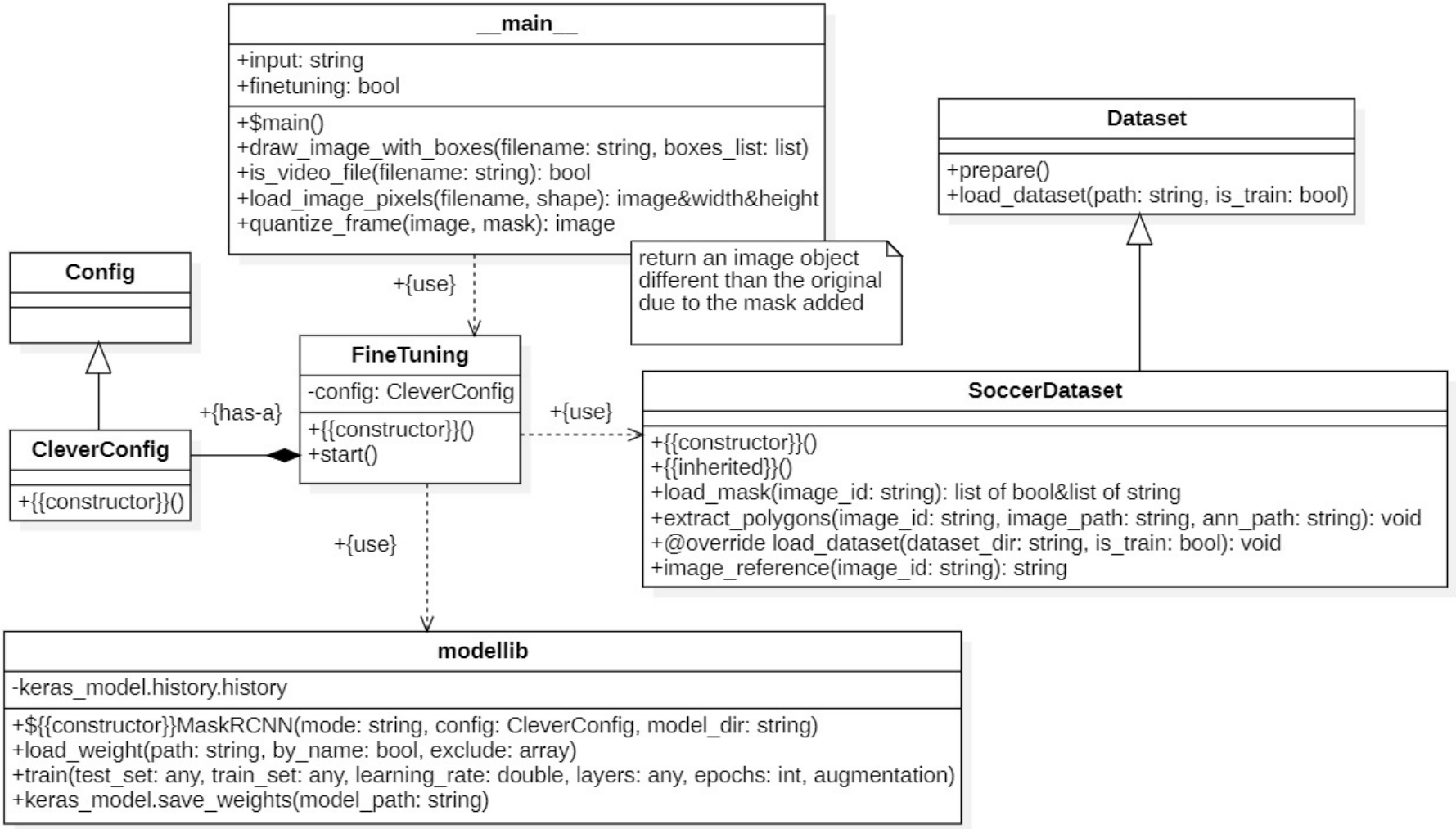


- Esempio di come la rete Mask-RCNN effettua la segmentazione(instance segmentation): in evidenza le RoI dei giocatori a cui sarà tolta un'informazione trascurabile, mentre le regioni di non interesse saranno quantizzate e quindi private delle alte frequenze.

- Quantizzazione con LPF + quantizzazione Codec a CRF costante

Schema tecnico

Diagramma della soluzione adottata



Fine tuning

Frammenti schema tecnico

Il modulo model, chiamato nel codice con alias modellib, è la classe wrapper fondamentale dove risiede il costruttore della rete e tutti i metodi ad essa associati per effettuare il fine tuning.

- Caricamento degli weights: connessioni tra i layers della rete per mezzo del quale essa effettua la detection e la segmentazione, in essi ci sono le regole per una corretta rilevazione delle entità specificate nei file di annotazione. Il caricamento degli weights è limitato a tutti i layers meno gli ultimi 4, che riconoscono 60 entità standard mentre rimuovendo questi layers noi ci assicuriamo che trovi solo calciatori e palla.

- Fine tuning rete preaddestrata, 2 passi: addestramento dei top layers (**heads**) "scongelamento" e addestramento dei layers della rete ResNet101, dal livello 4 in su(**4+**).

- **Augmentation:** i frame vengono analizzati dalla rete neurale dopo essere stati modificati con uno o più effetti tra i quali il ribaltamento rispetto ad un asse, la rotazione di 90,180 e 270 gradi, un leggero LPF gaussiano. Questo previene che i frame analizzati di qualsiasi video al di fuori del nostro dataset forniscano una segmentazione insoddisfacente.

Metriche di qualità

SSIM e LPIPS

SSIM: Structural Similarity Index
- Metrica con riferimento basata sulla somiglianza strutturale di due immagini.
- Separa il confronto tra struttura, luminanza e contrasto.
- Applicata spesso solo su luminanza di blocchi di campioni.
- Valori da -1 a 1: **1** quando l'immagine è perfettamente equivalente a quella di riferimento.
- Evoluzione: **MS-SSIM**
- Differenza: effettua varie fasi di sottocampionamento.

LPIPS: Learned Perceptual Image Patch Similarity
- Metrica con riferimento basata su CNN che effettuano image classification.
- Basata sul concetto di distanza tra immagini secondo l'occhio umano (perceptual loss)
- Generalmente migliore di **SSIM**
- Debole contro gli adversarial attack: attacchi basati sull'aggiunta di rumore in un frame grazie al quale si può far riconoscere un oggetto all'occhio umano invisibile.
- Evoluzione:**E-LPIPS**
- Differenze: utilizza un **augmentation** precedentemente al paragone delle due immagini.

Paragoni su risultati della codifica

Comparazione tra qualità di frame e maschere di video smart e standard

Il risultato della codifica dei 5 video di test descritti in seguito è stato analizzato con la metrica di qualità SSIM e LPIPS concludendo con i risultati seguenti:

	Qualità frame	Qualità maschere	Bitrate
Video smart (Fiorentina-Roma)	SSIM: 0,0358 LPIPS:170,45	SSIM: 0,0023 LPIPS:165,97	10,8 Mbit/s
Video standard (Fiorentina-Roma)	SSIM: 0,0343 LPIPS:173,32	SSIM: 0,0023 LPIPS:167,60	20,6 Mbit/s
Video smart (Chievo-Napoli)	SSIM: 0,14 LPIPS:173,44	SSIM: 0,0126 LPIPS:165,76	3,84 Mbit/s
Video standard (Chievo-Napoli)	SSIM: 0,1377 LPIPS:174,53	SSIM: 0,0126 LPIPS:167,761	6,2 Mbit/s
Video smart (Genoa-Milan)	SSIM: 0,1035 LPIPS:174,19	SSIM: 0,0483 LPIPS:164,29	8,9 Mbit/s
Video standard (Genoa-Milan)	SSIM: 0,1008 LPIPS:177,78	SSIM: 0,0483 LPIPS:168,13	17,7 Mbit/s

	Qualità frame	Qualità maschere	Bitrate
Video smart (Palermo-2Tempo)	SSIM: 0,0632 LPIPS:173,67	SSIM: 0,0632 LPIPS: 166,13	10,5 Mbit/s
Video standard (Palermo-2Tempo)	SSIM: 0,0623 LPIPS: 175,074	SSIM: 0,0623 LPIPS: 167,13	20,2 Mbit/s
Video smart (Napoli-Crotone)	SSIM: 0,7604 LPIPS: 167,31	SSIM: 0,7604 LPIPS: 164,50	15,7 Mbit/s
Video standard (Napoli-Crotone)	SSIM: 0,7551 LPIPS: 170,32	SSIM: 0,7551 LPIPS: 168,35	28,6 Mbit/s