CVID - Cours 2

Nicolas Boutry[1]

↪ nicolas.boutry@lrde.epita.fr

[1] Laboratoire de Recherche et Développement de l'EPITA (LRDE), France

Février 2023

Outline

Outline

Overview of predictive coding I

- Predictive coding is an important technique for image and video coding.
- In fact, temporal predictive coding using motion compensated prediction is the key of the success of video coding standards in 2000.
- The encoder must repeat the same process as the decoder to reproduce reconstructed samples; this is called closed-loop prediction.
- This kind of coder is generally referred to as differential pulse coded modulation (DPCM).

Overview of predictive coding II

- Error analysis when we send the predicted image plus the quantized prediction error image:
  - s original sample value,
  - s_p predicted sample value,
  - $e_p = s - s_p$ the original prediction error,
  - $\hat{e}_p$ the quantized prediction error,
  - $e_q = e_p - \hat{e}_p$ the quantization error for e,
  - the reconstruction $\hat{s}$ for s is:

$$\begin{aligned} \hat{s} &= s_p + \hat{e}_p && (1)\\ &= s_p + e_p - e_q && (2)\\ &= s_p + s - s_p - e_q && (3)\\ &= s - e_q, && (4) \end{aligned}$$

Overview of predictive coding III

- Therefore, the error between the original and the reconstructed sample value is:

$$s - \hat{s} = e_q,$$

exactly the same as the quantization error for the prediction error,

- Thus, the distortion in a lossy predictive coder is completely dependent on the quantizer for the prediction error, for a fixed predictor.

Motion compensated temporal prediction (unidirectional) I

- Uni-directional temporal prediction: we predict a pixel value in the current frame from its corresponding pixel in a previous frame.
- Let $\psi(x, t)$ represent the pixel value in frame t at pixel x, and let t^{-} denote the previous frame time. When the prediction process is described by:

$$\psi_p(x, t) = \psi(x, t_{-}),$$

  this is known as linear temporal prediction.
- Note: such type of prediction is effective only if the underlying scene is stationary.
- In a real-world video, the objects in the scene as well as the camera are usually moving.
- In this case, motion-compensated prediction (MCP) is more appropriate, which uses:

$$\psi_p(x, t) = \psi(x + d(x), t_{-}),$$

  where $d(x)$ represent the motion vector (MV) of pixel x from time t to time t_{-}.

Motion compensated temporal prediction (unidirectional) II

- Recall: frame $\psi(x, t)$ is the current frame, and frame $\psi(x, t_-)$ is the reference frame, and $\psi_p(x, t)$ is the predicted frame.
- Remember: the reference frame must be coded and reconstructed before the current frame.
- Theoretically, using pixels from more than one previous frame can improve prediction accuracy.

Motion compensated temporal prediction (bidirectional) I

- In bidirectional temporal prediction, a pixel in a current frame is predicted from a pixel in a previous frame t_- as well as a pixel in a following frame t_+.
- The predicted value at frame t is described by:

$$\psi_p(x,t) = a^- \; \psi(x + d^-(x), t_-) + a^+ \; \psi(x + d^+(x), t^+),$$

where $d^-(x)$ and $d^+(x)$ represent the MV at x from t to t^- and that from t to t^+.

- Typically, we call:
  - the prediction of the current frame from a previous ($t_- < t$) reference frame forward motion compensation,
  - the prediction of the current frame from a future ($t^+ > t$) reference frame backward motion compensation.

Motion compensated temporal prediction (bidirectional) II

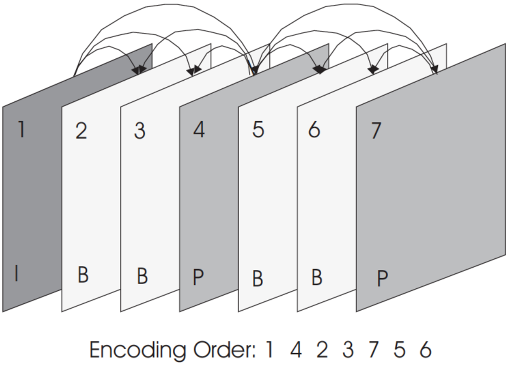


Figure 9.12. Video coding using both uni-directional and bi-directional temporal prediction. The arrows indicate the reference frames used for predicting a coded frame. Frames labeled I, P, and B are coded without prediction, with uni-directional prediction, and with bi-directional prediction, respectively.

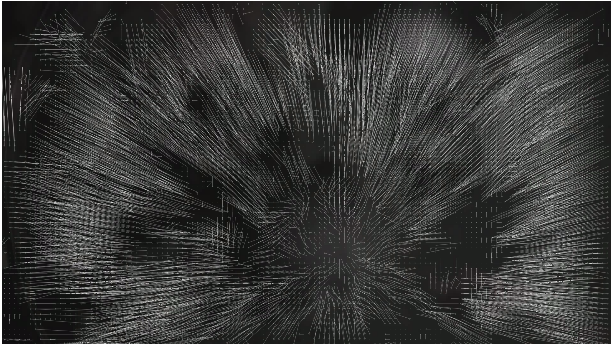
Motion compensated temporal prediction (bidirectional) III

- Note that the use of bi-directional prediction needs the coding of frames in an order that is different from the original temporal order.
- Bi-directional prediction implies encoding delay and is typically not used in real-time applications (video phone or video conferencing).
- The MPEG standard series, targeted mainly for video distribution, employ both uni- and bi-directional prediction.

Outline

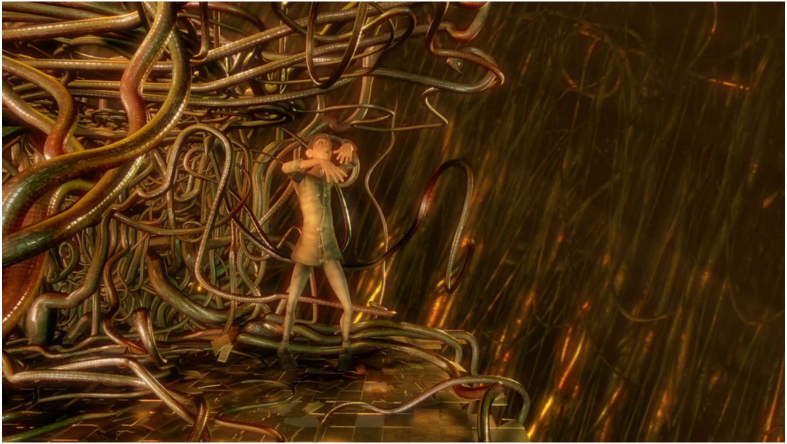
Motion estimation I

- Motion estimation (ME) is the process of determining motion vectors that describe the transformation from one 2D image to another.
- It is an ill-posed problem as the motion is in three dimensions but the images are a projection of the 3D scene onto a 2D plane.



Motion Compensation I

- Motion compensation (MC) is an algorithmic technique based on ME used to predict a frame in a video, given the previous and/or future frames.



Motion Compensation II

Figure: A frame ϕ_1 extracted from a video.

Motion Compensation III

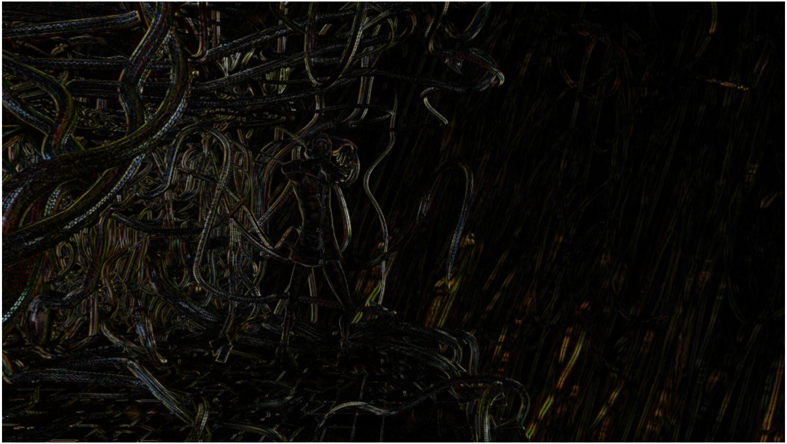


Figure: The difference between ϕ_1 and its following frame ϕ_2.

Motion Compensation IV

Figure: The difference between the MC-predicted frame $\hat{\phi}_2$ and ϕ_2 (more efficient !!!).

Motion Compensation V

- When we want to encode two consecutive frames ϕ_1 (reference) and ϕ_2 (current) using MC, we will then have to encode (for example):
  - The first frame ϕ_1 (using DCT),
  - The MVs predicting ϕ_2 from ϕ_1,
  - The prediction error $\varepsilon = \phi_2 - \hat{\phi}_2$
- Note: the reference picture may be previous in time or even from the future.
- Most video coding standards (H.26x, MPEGs) use motion-compensated DCT video coding (block motion compensation).

Outline

Introduction I

- Scalability is the capability of recovering physically meaningful image or video information from partial compressed bitstreams,
- Example: we would that an user with high bandwidth connection can download the entire bitstream to view the full quality video, while the user with a low-bandwidth connection will only download a subset of the stream, and see a low quality video.
- Scalable coders can have coarse granularity (two or three layers), or fine granularity,
- In the extreme case of fine granularity, the bit stream can be truncated at any point.
  - The more bits are retained, the better will be the reconstructed image quality.
  - We call such type of bitstream embedded.

Introduction II

- Scalable coding is typically accomplished by providing multiple versions of a video either in terms of:
  - amplitude resolutions (called quality scalability or SNR scalability),
  - temporal resolutions (temporal scalability),
  - frequency resolution (frequency scalability),
  - a combination of these options.
- When scalable contents can be accessed at object level, we call this object-based scalability (MPEG4).
- Simulcast simply codes the same video with different resolutions,
- This method is simple but not efficient: it encodes several times the same information.

Quality scalability I

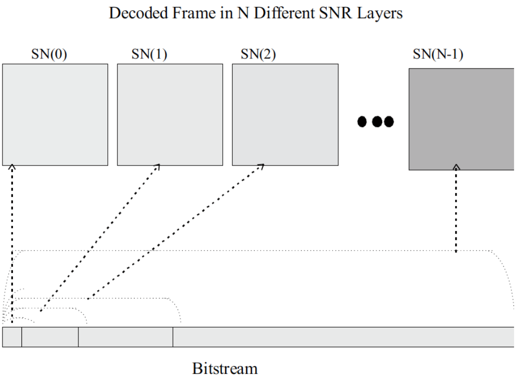


Figure: Quality scalability

Quality scalability II

- Decoding the first layer (also called base layer) provides a low quality version of the reconstructed image.
- Further decoding the remaining layers (also called enhancement layers) results in a quality increase of the reconstructed image up to the highest quality.
- The first layer is obtained by applying a coarse quantizer to the original image or in a transform (e.g., DCT) domain.
- The second layer contains the quantized difference between the original image and that reconstructed from the first layer,
- This quantizer that is finer than that used to produce the first layer.
- And we continue this way using increasingly finer quantizers.

Quality scalability III

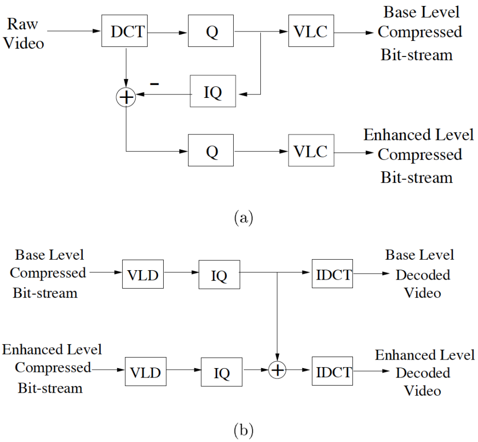


Figure 11.3. A two level quality-scalable codec. (a) encoder; (b) decoder.

Spatial scalability I

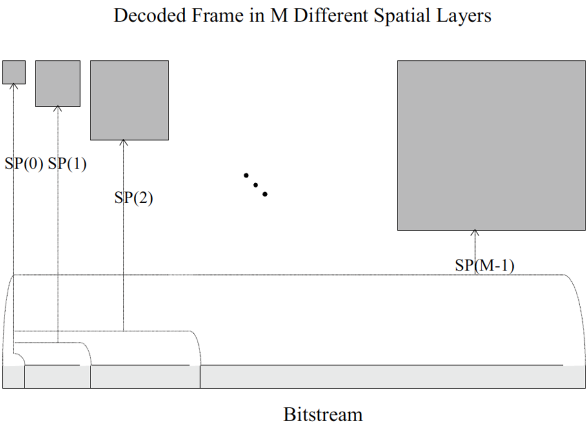


Figure: Spatial scalability

Spatial scalability II

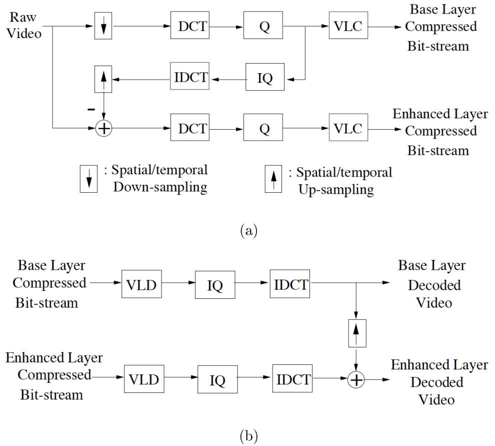


Figure 11.6. A two level spatially/temporally scalable codec. (a) encoder; (b) decoder.

Spatial scalability III

- Spatial scalability is defined as representing the same video in different spatial resolutions or sizes.
- By decoding the first layer, the user can display a preview version of the decoded image at a lower resolution.
- Decoding the second layer results in a larger reconstructed image.
- By progressively decoding the additional layers, the viewer can increase the spatial resolution of the image up to the full resolution of the original image.
- To produce such a layered bit stream, we must compute a multi-resolution decomposition of the original image.

Temporal scalability I

- Temporal scalability is defined as representing the same video in different temporal resolutions or frame rates.
- Temporal scalability enables different frame rates for different layers of the contents.
- The block diagram of temporally scalable codec is the same as that of spatially scalable codec.
- The simplest temporal down-sampling is by frame skipping.
- Temporal up-sampling can be accomplished by frame copying.
- Note that the reasoning is different in space and in time due to the different perceptions!

Frequency scalability I

- We include different frequency components in each layer.
- The base layer contains low frequency components,
- The other layers contain increasingly higher frequency components.
- This way, the base layer will provide a blurred version of the image, and the addition of enhancement layers will yield increasingly sharper images.
- Whole-frame transforms: subband decompositions, wavelet transforms.
- Block-based transforms: block DCT.

Combination of basic schemes I

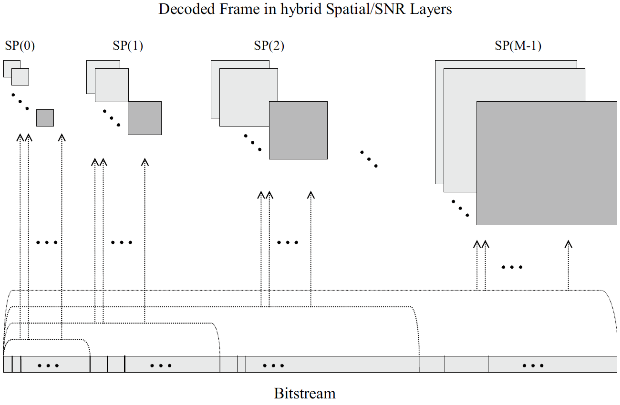


Figure 11.7. NxM layers of combined spatial/quality scalability. From [20, Fig. 3].

Combination of basic schemes II

- Quality, spatial, temporal and frequency scalabilities are basic scalable mechanisms.
- They can be combined to reach finer granularity.
- For example:

1. First we improve the image quality at a given spatial resolution,
2. Then we refine until the best quality is achieved at this spatial resolution,
3. Then we increase the spatial resolution to a higher level ... (and so on!)

Fine granularity scalability (FGS) I

- Fine granularity scalability refers to a coding method by which the rate as well as quality increment at a much smaller step (MPEG-4).
- When a bitstream can provide continuously improving video quality with every additional bit, the underlying coding method is called embedded coding.
- Note: embedded implies FGS but not the contrary.
- Obviously, FGS and embedded coding can adapt to bandwidth variations in real networks more effectively.
- In practice, a base layer is first produced to provide a low but guaranteed level of quality,
- Then an enhancement layer may be generated to provide improvement in fine granularity.

Object-based scalability I

- Object temporal scalability: the frame rate of the object is higher that the one of the remaining area.

Object-based scalability II

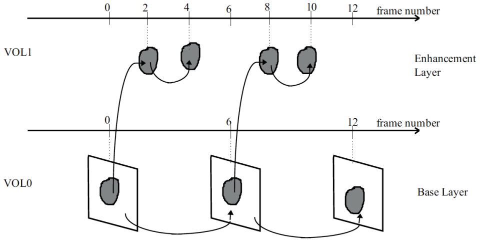


Figure 11.9. Enhancement structure of Type 1 with P-VOPs (Courtsey of MPEG4)

$\rightsquigarrow$ More information to encode the object compared to the background .

Object-based scalability III

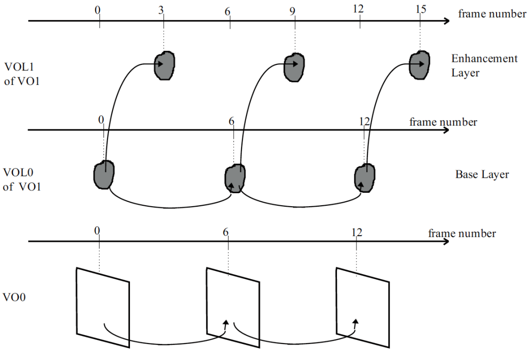


Figure 11.11. Enhancement structure of Type 2 (Courtesy of MPEG4)

Wavelet transform based coding I

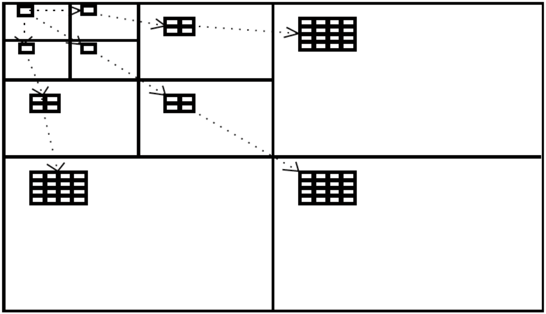


Figure 11.12. The parent-child relationship of wavelet coefficients. From [20, Fig. 4].

⇝ LL, HL1, LH1, HH1, HL2, LH2, HH2, HL3, LH3, HH3.

Wavelet transform based coding II

- The discrete wavelet transform (DWT) provides a mutliresolution/multifrequency expression of a signal with localization in both time and frequency.
- The multiresolution/multifrequency decomposition offered by the wavelet transform lends itself easily to a scalable bit stream.
- Like DCT-based approach, wavelet transform based coding for images consists of threes steps:
  - (1) wavelet transform,
  - (2) quantization,
  - (3) entropy coding.
- The results in matter of compression are relatively the same as the MPEG-4's DCT-based coder (in terms of PSNR).
- At the end, it has been shown that the optimization of the whole framework (coding, ...) is more important that optimizing the transform itself.

Outline

Overview of a video coding system I

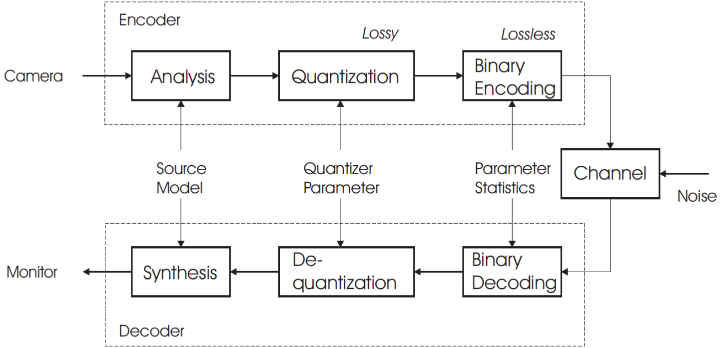


Figure 8.1. Overview of a video coding system.

References

Indrajit Chakrabarti, Kota Naga Srinivasarao Batta, S. K. C. (2015).
Motion Estimation for Video Coding.
Springer.

Jie Chen, Ut-Va Koc, K. R. L. (2002).
Design of Digital Video Coding Systems.
Signal Processing and Communication Series, Marcel Dekker, Inc.

Yao Wang, Jorn Ostermann, Y.-Q. Z. (2002).
Video Processing and Communications.
Prentice Hall.