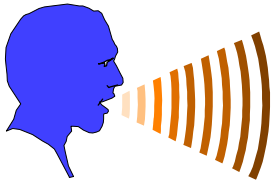


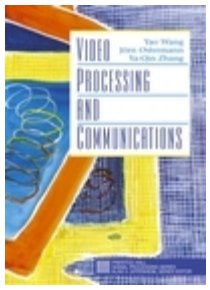



Video Processing & Communications

Video Coding Using Motion Compensation

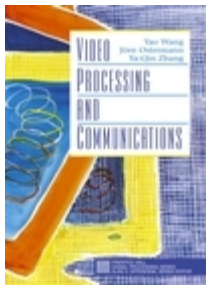
Yao Wang
Polytechnic University, Brooklyn, NY11201
http://eeweb.poly.edu/~yao

Based on: Y. Wang, J. Ostermann, and Y.-Q. Zhang, Video Processing and Communications, Prentice Hall, 2002.

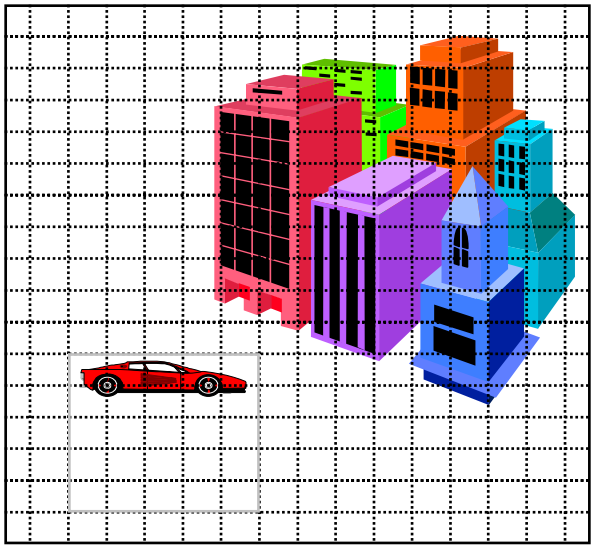


Outline

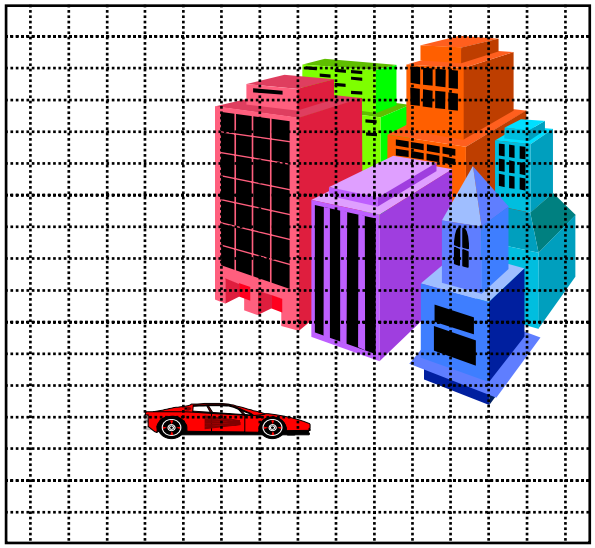
- Block-Based Hybrid Video Coding
  - Overview of Block-Based Hybrid Video Coding
  - Overlapped Block Motion Compensation
  - Coding mode selection and rate control
  - Loop filtering
- Scalable Video Coding
  - Motivation for scalable coding
  - Basic modes of scalability
  - Scalability in MPEG-2
  - Fine granularity scalability in MPEG-4



Characteristics of Typical Videos

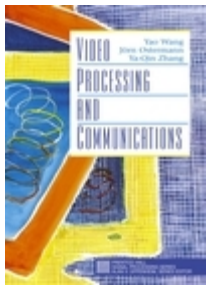


Frame *t-1*



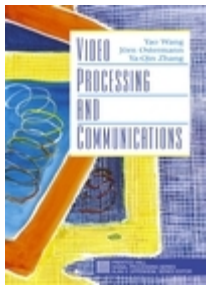
Frame *t*

Adjacent frames are similar and changes are due to object or camera motion

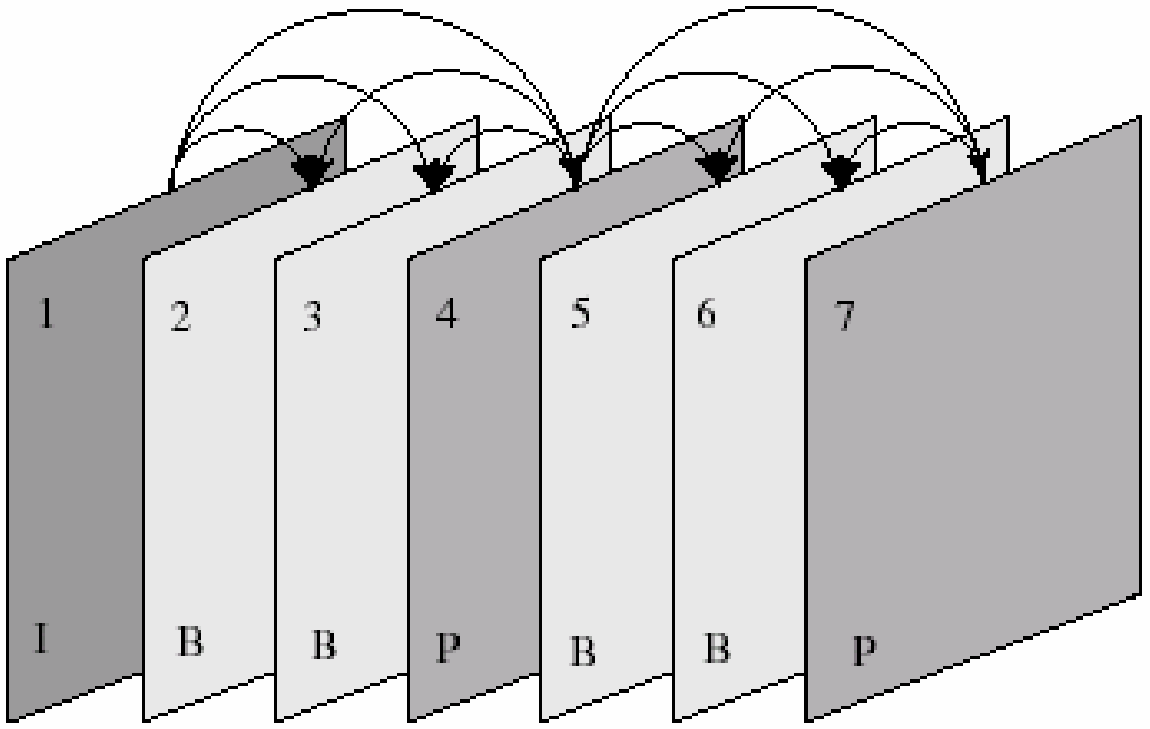


Key Idea in Video Compression

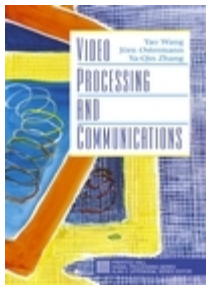
- Predict a new frame from a previous frame and only code the prediction error --- Inter prediction
- Predict a current block from previously coded blocks in the same frame --- Intra prediction (introduced in the latest standard H.264)
- Prediction error will be coded using the DCT method
- Prediction errors have smaller energy than the original pixel values and can be coded with fewer bits
- Those regions that cannot be predicted well will be coded directly using DCT --- Intra coding without intra-prediction
- Work on each macroblock (MB) (16x16 pixels) independently for reduced complexity
  - Motion compensation done at the MB level
  - DCT coding of error at the block level (8x8 pixels)



Different Coding Modes



Encoding order: 1 4 2 3 7 5 6



Temporal Prediction

- No Motion Compensation:
  - Work well in stationary regions

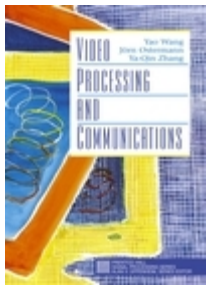
$$\hat{f}(t,m,n) = f(t-1,m,n)$$

- Uni-directional Motion Compensation:
  - Does not work well for uncovered regions by object motion

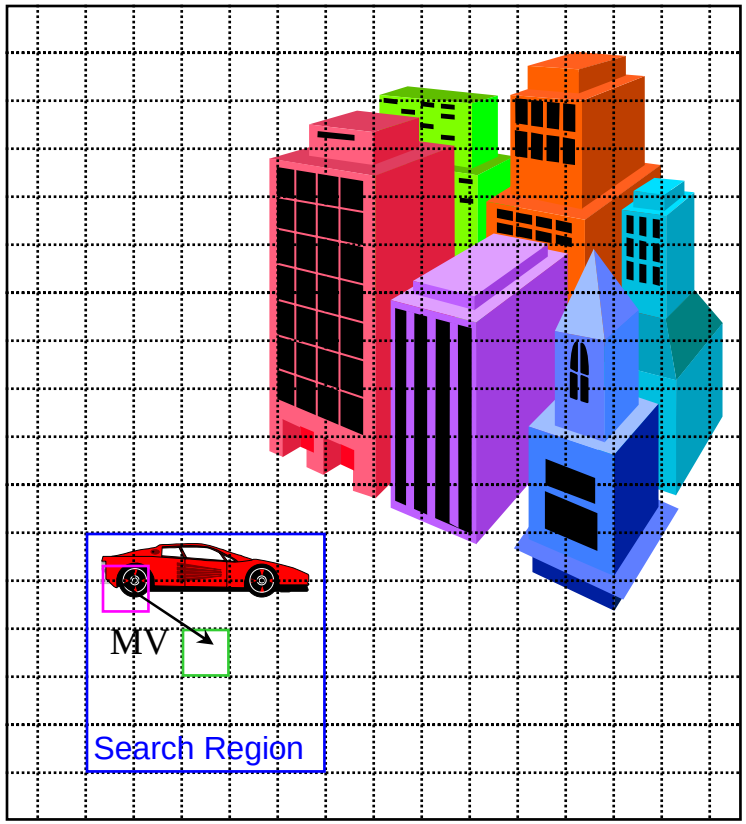
$$\hat{f}(t,m,n) = f(t-1,m-d_x,n-d_y)$$

- Bi-directional Motion Compensation
  - Can handle better uncovered regions

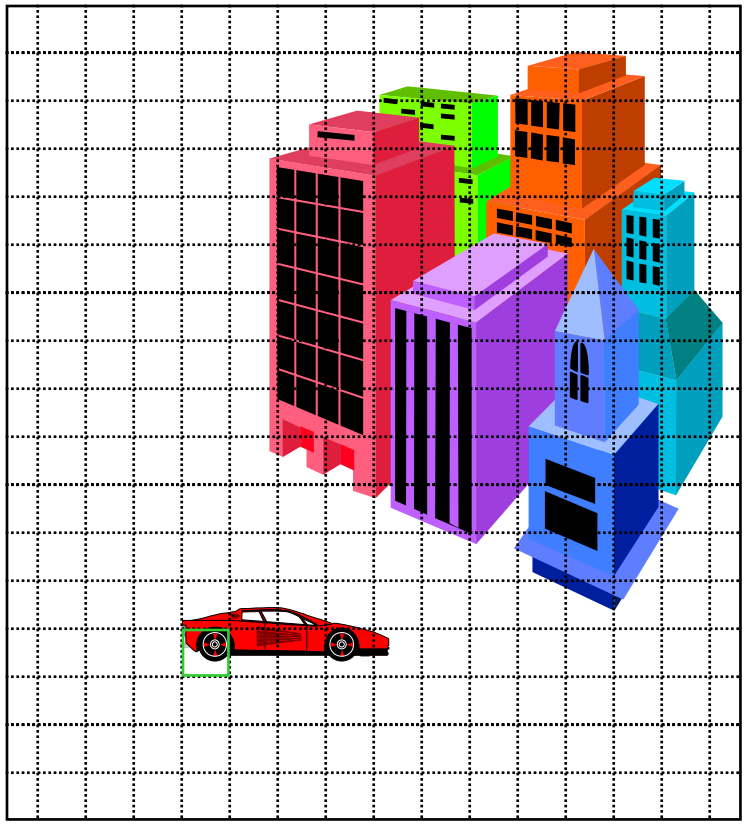
$$\hat{f}(t,m,n) = w_b f(t-1,m-d_{b,x},n-d_{b,y}) + w_f f(t+1,m-d_{f,x},n-d_{f,y})$$



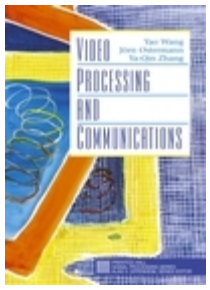
Block Matching Algorithm for Motion Estimation



Frame *t-1*
(Reference Frame)



Frame *t*
(Predicted frame)



Multiple Reference Frame Temporal Prediction

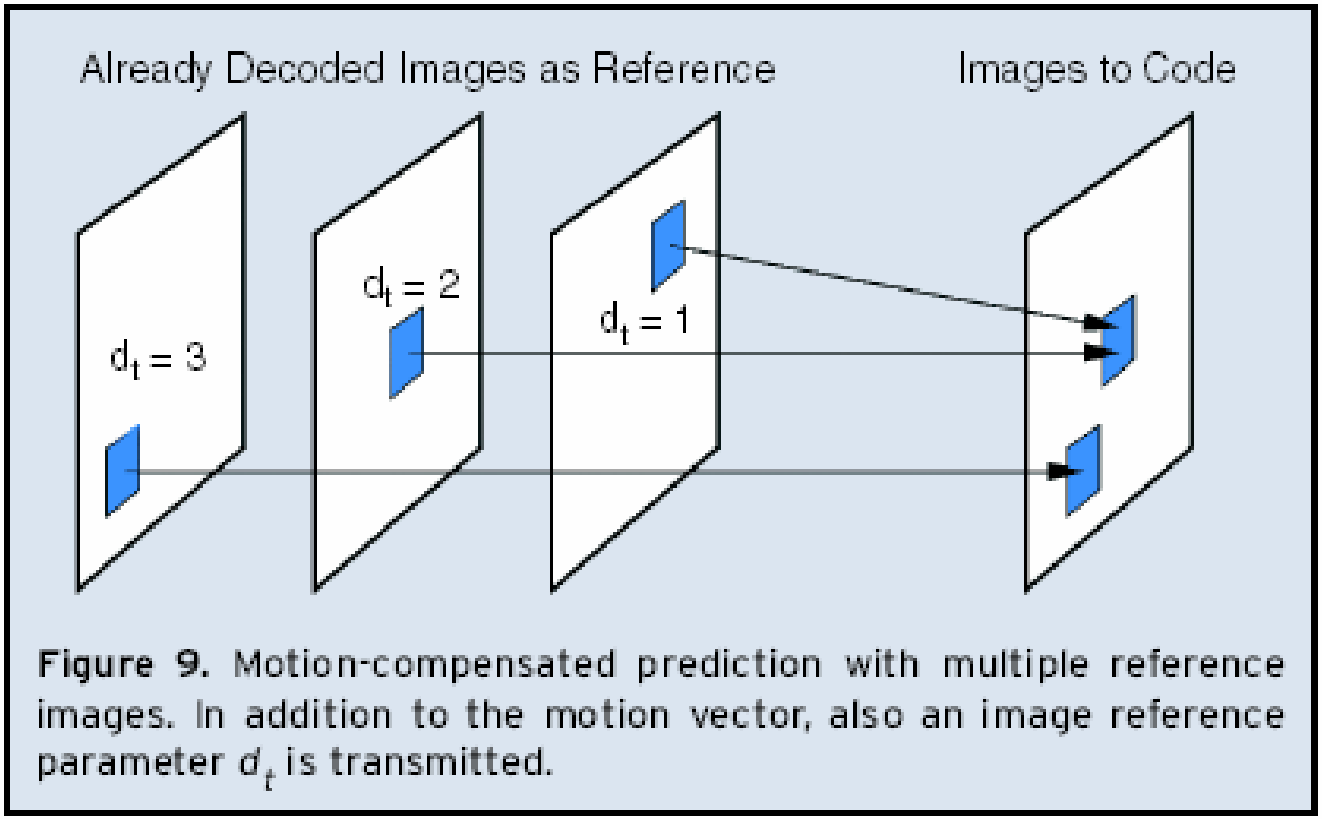
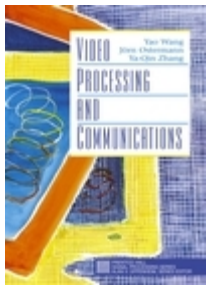


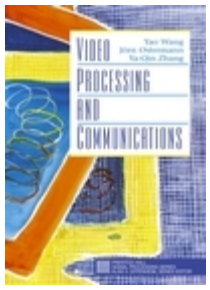
Figure 9. Motion-compensated prediction with multiple reference images. In addition to the motion vector, also an image reference parameter d_t is transmitted.

When multiple references are combined, the best weighting coefficients can be determined using ideas similar to minimal mean square error predictor



Spatial Prediction

- General idea:
  - A pixel in the new block is predicted from previously coded pixels in the same frame
  - What neighbors to use?
  - What weighting coefficients to use?
- Content-adaptive prediction
  - No edges: use all neighbors
  - With edges: use neighbors along the same direction
  - The best possible prediction pattern can be chosen from a set of candidates, similar to search for best matching block for inter-prediction
    - H.264 has many possible intra-prediction pattern



H.264 Intra-Prediction

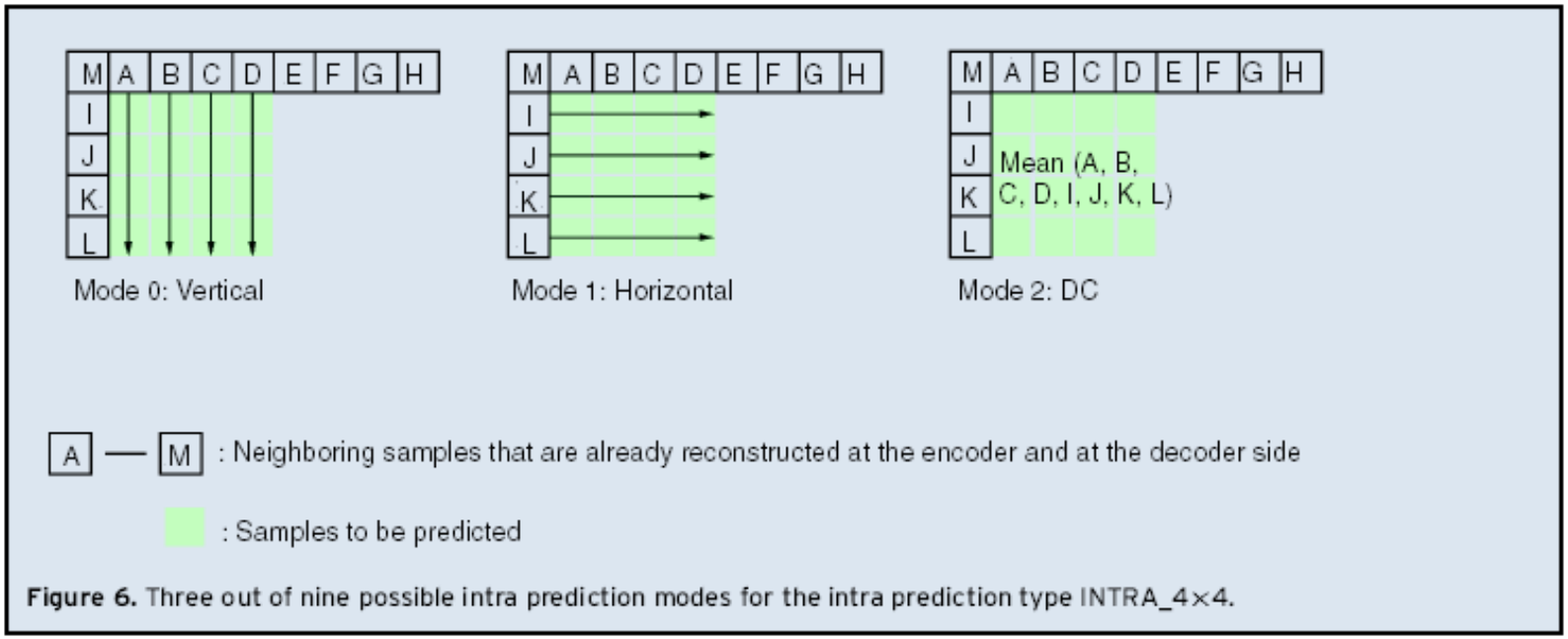
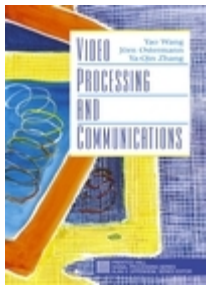
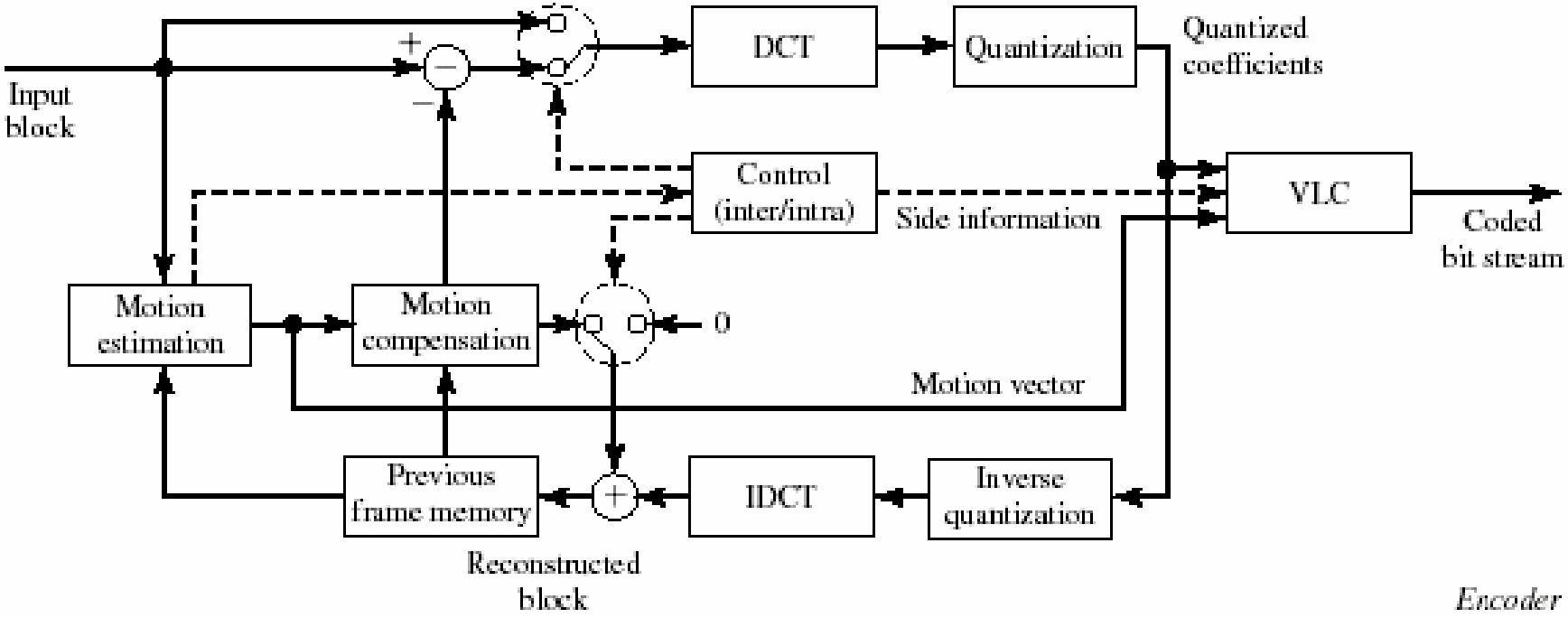


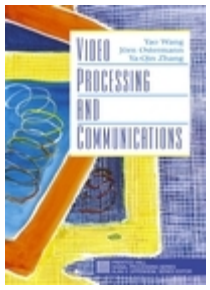
Figure 6. Three out of nine possible intra prediction modes for the intra prediction type INTRA_4×4.

From: Ostermann et al., Video coding with H.264/AVC: Tools, performance, and complexity, IEEE Circuits and Systems Magazine, First Quarter, 2004

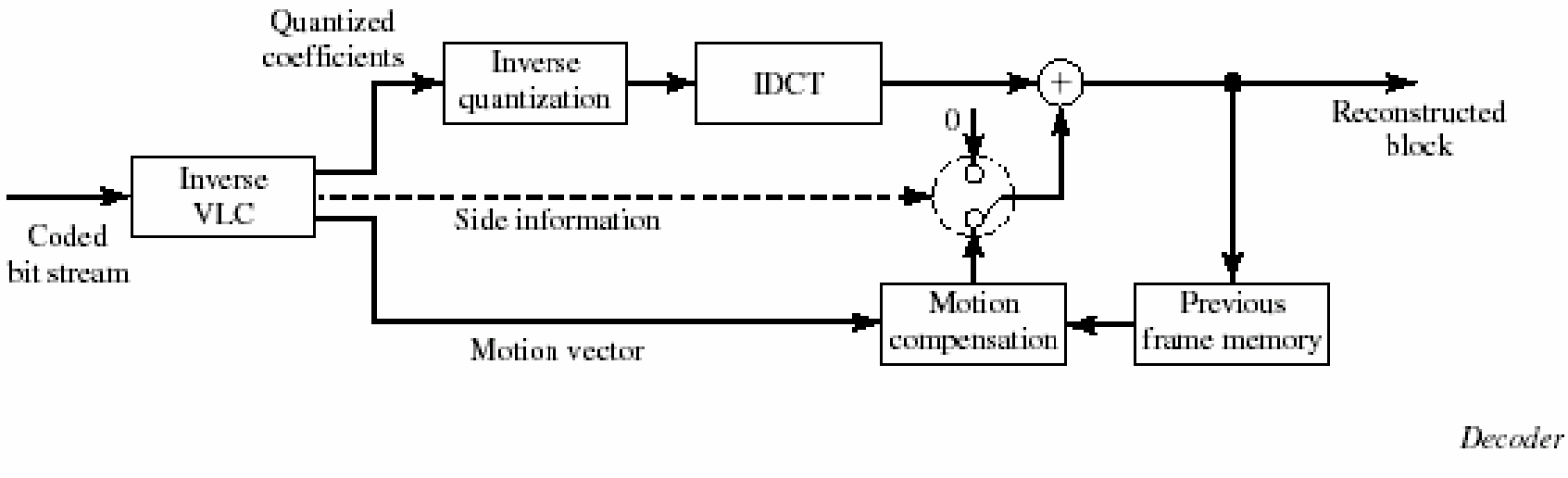


Encoder Block Diagram of a Typical Block-Based Video Coder (Assuming No Intra Prediction)

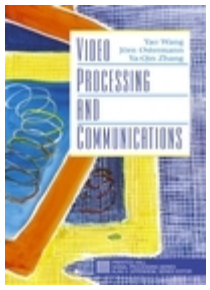




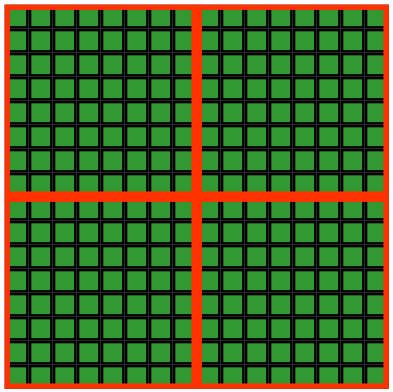
Decoder Block Diagram



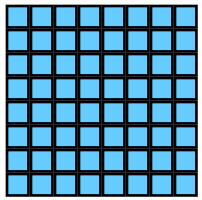
Decoder



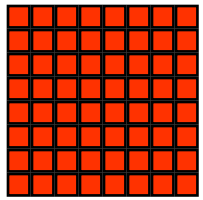
MB Structure in 4:2:0 Color Format



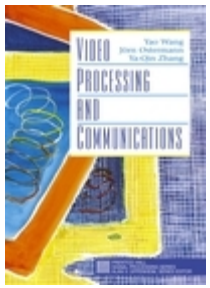
4 8x8 Y blocks



1 8x8 Cb blocks



1 8x8 Cr blocks



Macroblock Coding in I-Mode (assuming no intra-prediction)

DCT transform each 8x8 DCT block

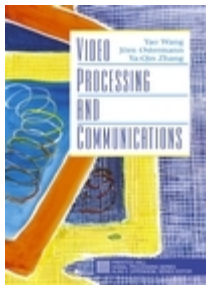
↓

Quantize the DCT coefficients with properly chosen quantization matrices

↓

The quantized DCT coefficients are zig-zag ordered and run-length coded

With intra-prediction, after the best intra-prediction pattern is found, the prediction error block is coded using DCT as above.



Macroblock Coding in P-Mode

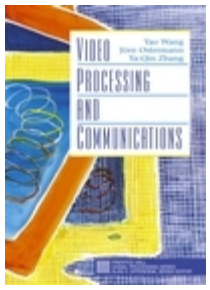
Estimate one MV for each macroblock (16x16)

↓

Depending on the motion compensation error, determine the coding mode (intra, inter-with-no-MC, inter-with-MC, etc.)

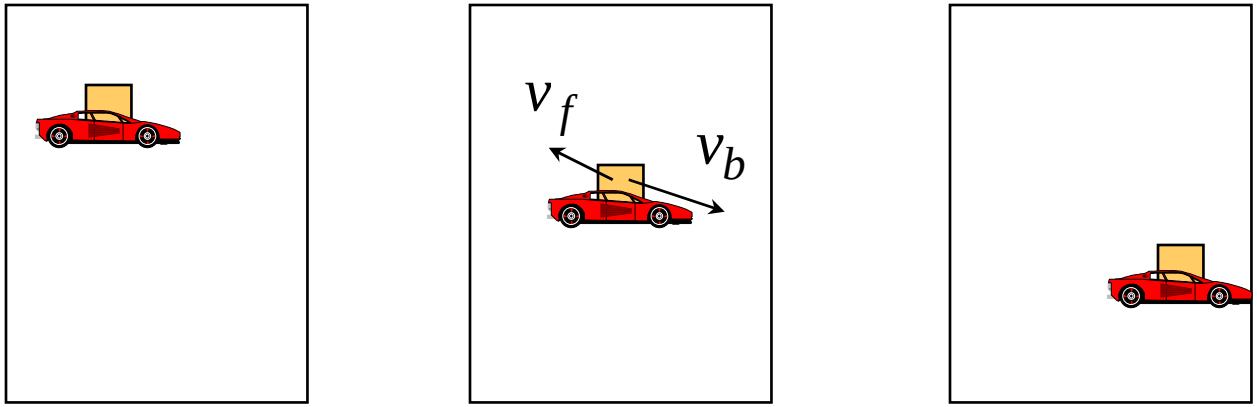
↓

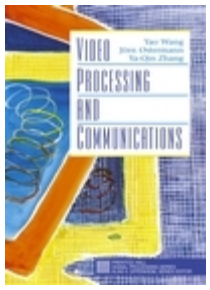
The original values (for intra mode) or motion compensation errors (for inter mode) in each of the DCT blocks (8x8) are DCT transformed, quantized, zig-zag/alternate scanned, and run-length coded



Macroblock Coding in B-Mode

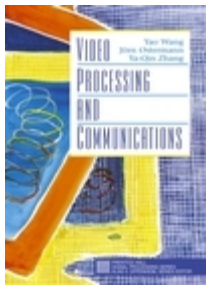
- Same as for the P-mode, except a macroblock can be predicted from a previous picture, a following one, or both.



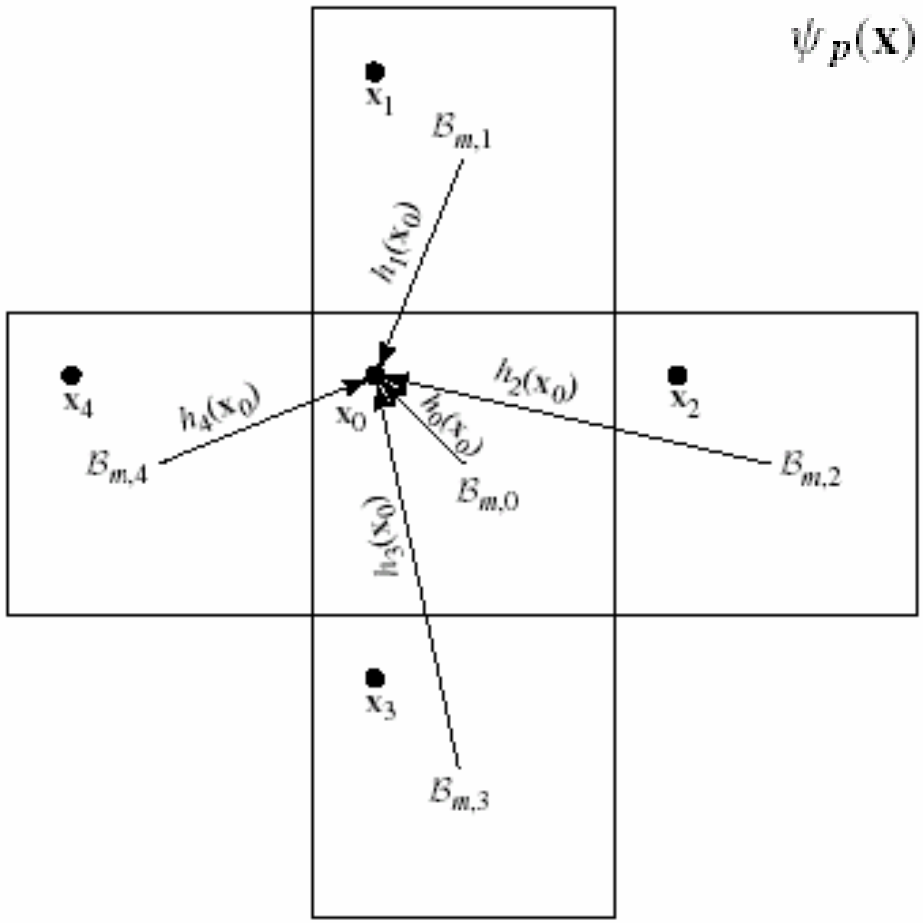


Overlapped Block Motion Compensation (OBMC)

- Conventional block motion compensation
  - One best matching block is found from a reference frame
  - The current block is replaced by the best matching block
- OBMC
  - Each pixel in the current block is predicted by a weighted average of several corresponding pixels in the reference frame
  - The corresponding pixels are determined by the MVs of the current as well as adjacent MBs
  - The weights for each corresponding pixel depends on the expected accuracy of the associated MV

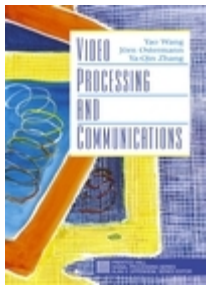


OBMC Using 4 Neighboring MBs



$$\psi_p(\mathbf{x}) = \sum_{k \in \mathcal{K}} h_k(\mathbf{x}) \psi_r(\mathbf{x} + \mathbf{d}_{m,k}), \quad \mathbf{x} \in \mathcal{B}_m.$$

$h_k(\mathbf{x})$ Should be inversely proportional to the distance between x and the center of $\mathcal{B}_{m,k}$.



Weighting Coefficients Used in H.263

4	5	5	5	5	5	5	4
5	5	5	5	5	5	5	5
5	5	6	6	6	6	5	5
5	5	6	6	6	6	5	5
5	5	6	6	6	6	5	5
5	5	6	6	6	6	5	5
5	5	5	5	5	5	5	5
4	5	5	5	5	5	5	4

(a)

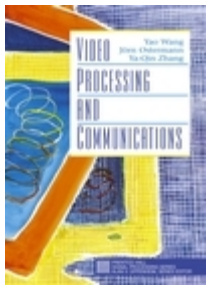
2	2	2	2	2	2	2	2
1	1	2	2	2	2	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	2	2	2	2	1	1
2	2	2	2	2	2	2	2

(b)

2	1	1	1	1	1	1	2
2	2	1	1	1	1	2	2
2	2	1	1	1	1	2	2
2	2	1	1	1	1	2	2
2	2	1	1	1	1	2	2
2	2	1	1	1	1	2	2
2	2	1	1	1	1	2	2
2	1	1	1	1	1	1	2

(c)

Figure 9.16 (a) The weighting coefficients for OBMC, specified in the H.263 video coding standard [17]: (a) for prediction with the motion vector of the current block; (b) for prediction with motion vectors of the blocks on top or bottom of the current block; (c) for prediction with motion vectors of the blocks to the left or right of the current block. The numbers given are 8 × the actual weights. For example, to predict the pixel at the top left corner of the block, the weights associated with the MVs of the current MB, the top MB, and the left MB are 4/8, 2/8, and 2/8, respectively. For the pixel on the first row and the second column, the weights are 5/8, 2/8, 1/8, respectively.



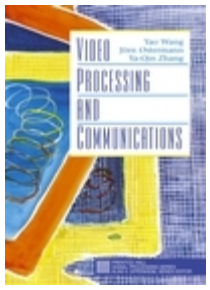
Optimal Weighting Design

- Convert to an optimization problem:
  - Minimize $$E\left\{\left|\psi(\mathbf{x})-\sum_{k\in\mathcal{K}}h_k(\mathbf{x})\psi_r(\mathbf{x}+\mathbf{d}_{m,k})\right|^2\right\}.$$
  - Subject to $$\sum_{k\in\mathcal{K}}h_k(\mathbf{x})=1$$
- Optimal weighting functions:

$$\mathbf{h}(\mathbf{x})=[\mathbf{R}(\mathbf{x})]^{-1}\left(\mathbf{r}(\mathbf{x})-\frac{\mathbf{i}^T[\mathbf{R}(\mathbf{x})]^{-1}\mathbf{r}(\mathbf{x})-1}{\mathbf{i}^T[\mathbf{R}(\mathbf{x})]^{-1}\mathbf{i}}\mathbf{i}\right),$$

$$R_{k,l}(\mathbf{x})=E\{\psi_r(\mathbf{x}+\mathbf{d}_{m,k})\psi_r(\mathbf{x}+\mathbf{d}_{m,l})\},\ k,l\in\mathcal{K}$$

$$r_k(\mathbf{x})=E\{\psi(\mathbf{x})\psi_r(\mathbf{x}+\mathbf{d}_{m,k})\},\ k\in\mathcal{K}.$$



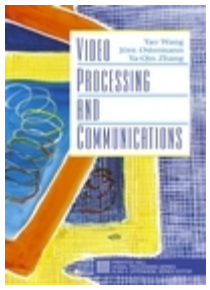
How to Determine MVs with OBMC

- Option 1: using conventional BMA, minimize the prediction error (MAD) within each MB independently
- Option 2: minimize the prediction error assuming OBMC
  - Solve the MV for the current MB while keeping the MVs for the neighboring MBs found in the previous iterations

$$\sum_{\mathbf{x}\in\mathcal{B}_m}\left|\psi(\mathbf{x})-\sum_{k\in\mathcal{K}}h_k(\mathbf{x})\psi_r(\mathbf{x}+\mathbf{d}_{m,k})\right|^p.$$

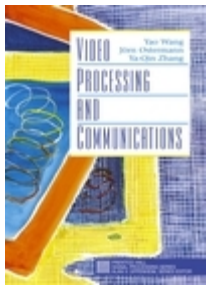
- Option 3: Using a weighted error criterion over a larger block

$$\sum_{k\in\mathcal{K}}\sum_{\mathbf{x}\in\mathcal{B}_{m,k}}|\psi(\mathbf{x})-\psi_r(\mathbf{x}+\mathbf{d}_m)|^p h_k(\mathbf{x}). \longrightarrow \sum_{\mathbf{x}\in\mathcal{B}_m}|\psi(\mathbf{x})-\psi_r(\mathbf{x}+\mathbf{d}_m)|^p\bar{h}(\mathbf{x})$$



Window Function Corresponding to H.263 Weights for OBMC

				1	1	1	1	1	1	1	1				
				1	1	1	1	1	1	1	1				
				1	1	2	2	2	2	1	1				
				2	2	2	2	2	2	2	2				
1	1	1	2	4	5	5	5	5	5	5	4	2	1	1	1
1	1	2	2	5	5	5	5	5	5	5	5	2	2	1	1
1	1	2	2	5	5	6	6	6	6	5	5	2	2	1	1
1	1	2	2	5	5	6	6	6	6	5	5	2	2	1	1
1	1	2	2	5	5	6	6	6	6	5	5	2	2	1	1
1	1	2	2	5	5	6	6	6	6	5	5	2	2	1	1
1	1	2	2	5	5	5	5	5	5	5	5	2	2	1	1
1	1	1	2	4	5	5	5	5	5	5	4	2	1	1	1
				2	2	2	2	2	2	2	2				
				1	1	2	2	2	2	1	1				
				1	1	1	1	1	1	1	1				
				1	1	1	1	1	1	1	1				



Coding Mode Selection

- Coding modes:
  - Intra vs. inter, QP to use, for each MB, each leading to different rate
- Rate-distortion optimized selection, given target rate:
  - Minimize the distortion, subject to the target rate constraint

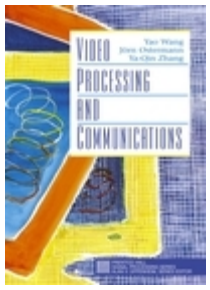
$$\text{minimize} \quad \sum_n D_n(m_n),$$
$$\text{subject to} \quad \sum_n R_n(m_k, \forall k) \le R_d.$$

$$\text{minimize} \quad J(m_n, \forall n) = \sum_n D_n(m_n) + \lambda \sum_n R_n(m_k, \forall k)$$

Simplified version $J_n(m_n) = D_n(m_n) + \lambda R_n(m_n).$

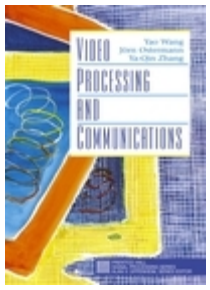
The optimal mode is chosen by coding the block with all candidates modes and taking the mode that yields the least cost.

Note that one can think of each candidate MV (and reference frame) as a possible mode, and determine the optimal MV (and reference frame) using this frame work --- Rate-distortion optimized motion estimation.



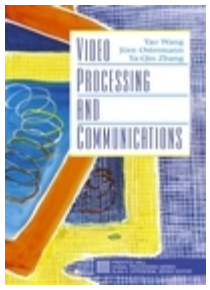
Rate Control

- Rate control:
  - The coding method necessarily yields variable bit rate
  - Rate control is necessary when the video is to be sent over a constant bit rate (CBR) channel, where the rate when averaged over a short period should be constant
  - The fluctuation within the period can be smoothed by a buffer at the encoder output
- Problem of rate control:
  - Step 1) Determine the target rate at the frame or GOB level, based on the current buffer fullness
  - Step 2) Satisfy the target rate by varying frame rate (skip frames when necessary) and QP
    - Determination of QP requires an accurate model relating rate with Q (quantization stepsize)
      - General model: R ~ A/Q + B/Q^2



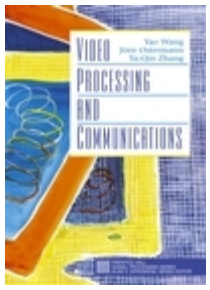
Loop Filtering

- Errors in previously reconstructed frames (mainly blocking artifacts) accumulate over time with motion compensated temporal prediction
  - Reduce prediction accuracy
  - Increase bit rate for coding new frames
- Loop filtering:
  - Filter the reference frame before using it for prediction
  - Can be embedded in the motion compensation loop
    - Half-pel motion compensation
    - OBMC
  - Explicit deblocking filtering: removing blocking artifacts after decoding each frame
- Loop filtering can significantly improve coding efficiency
- Theoretically optimal design of loop filters:
  - See text

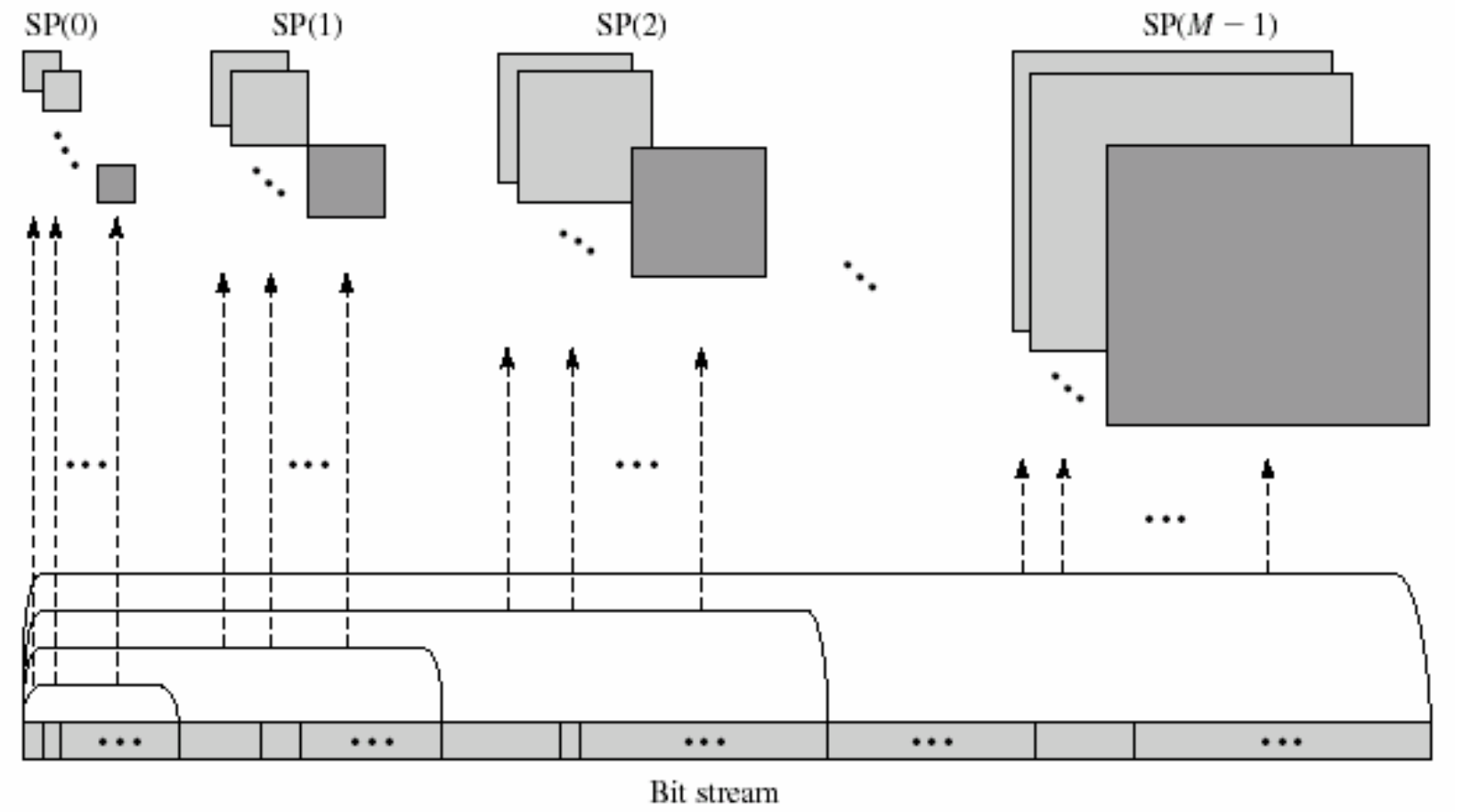


Scalable Coding

- Motivation
  - Real networks are heterogeneous in rate
  - The same video may be accessed by users with different access bandwidth and decoding/display capability
    - streaming video from a cell phone, from home (56 kbps) using modem vs. corporate LAN (10-100 mbps)
- Scalable video coding
  - Ideal goal (embedded stream): Creating a bitstream that can be accessed at any rate
  - Practical video coder:
    - layered coder: base layer provides basic quality, successive layers refine the quality incrementally
    - Fine granularity (FGS)



Bit Stream Scalability



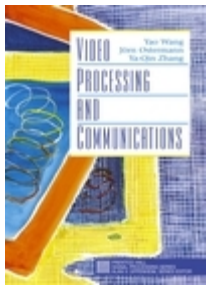


Illustration of Scalable Coding

6.5 kbps

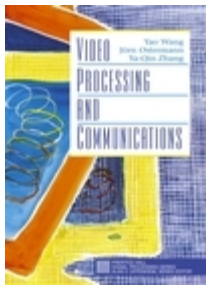
133.9 kbps

Spatial scalability

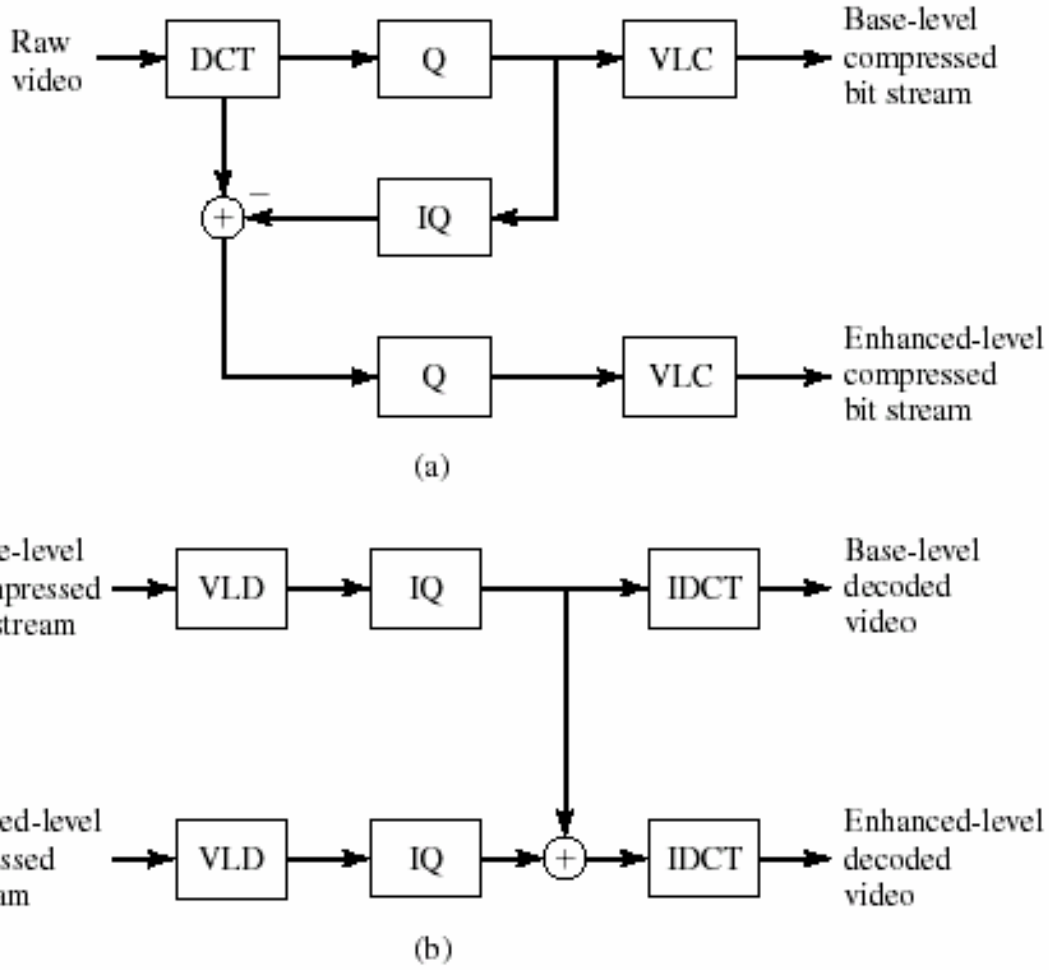
21.6 kbps

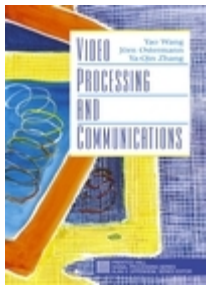
436.3 kbps

Quality (SNR) scalability

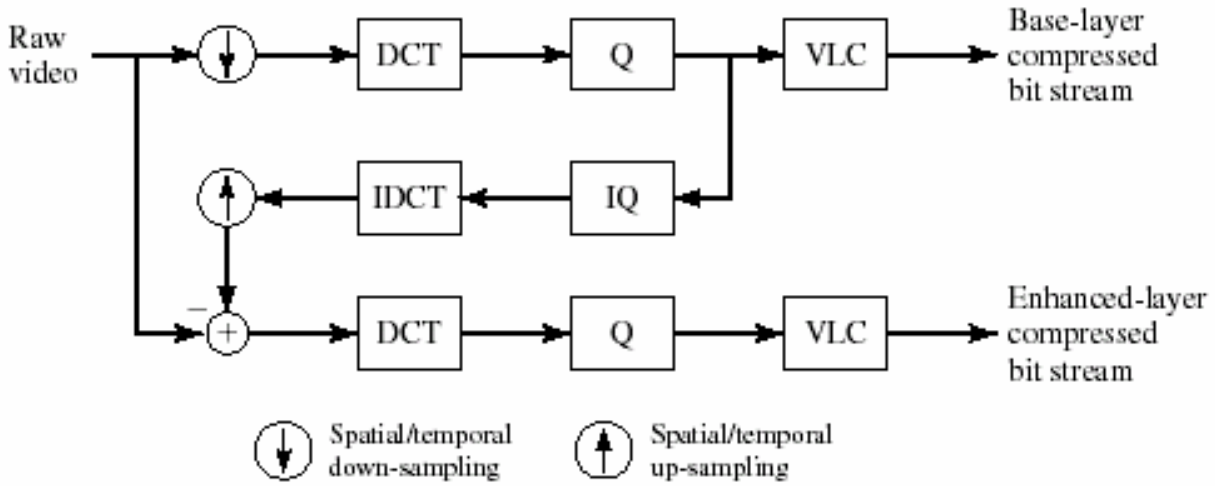


Quality (SNR) Scalability By Multistage Stage Quantization

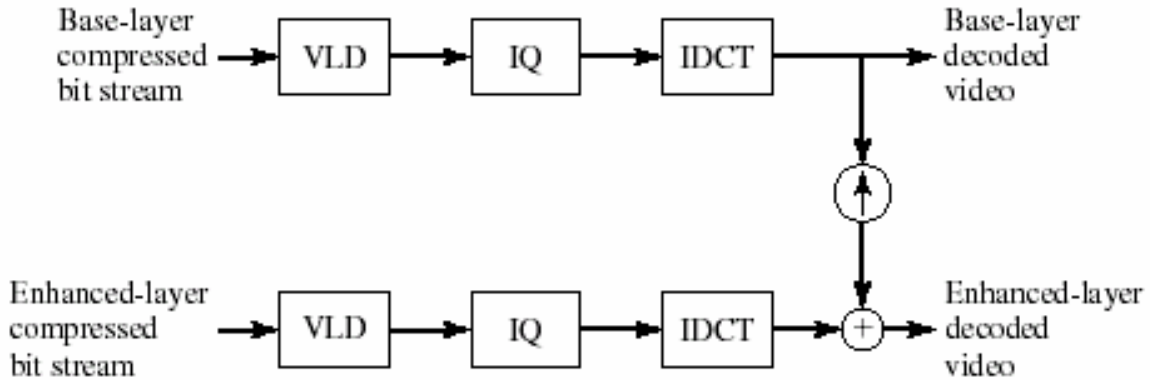




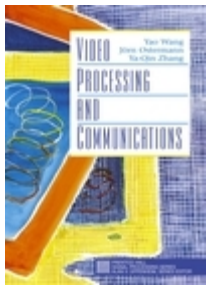
Spatial/Temporal Scalability Through Down/Up Sampling



(a)

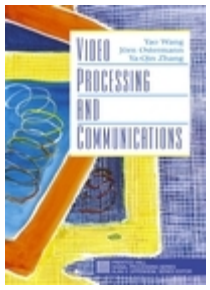


(b)



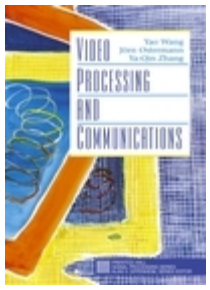
Scalability in MPEG-2

- MPEG-2 is the earliest standard that offers scalability tools
- Four types of scalability
  - Data partition (frequency scalability)
  - SNR scalability (quality scalability)
  - Temporal scalability (frame-rate scalability)
  - Spatial scalability (resolution scalability)



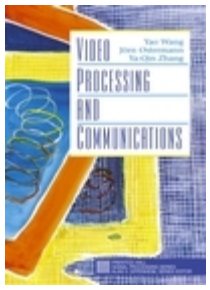
Fine Granularity Scalability in MPEG-4

- MPEG-4 achieves fine granularity quality scalability through bit-plane coding
  - The DCT coefficients are represented losslessly in binary bits
  - The bit planes are coded successively, from the most significant bit to the least.
  - The bit plane within each block is coded using run-length coding.
- Temporal scalability is accomplished by combining I, B, and P-frames
- Spatial scalability is achieved by spatial down/up sampling



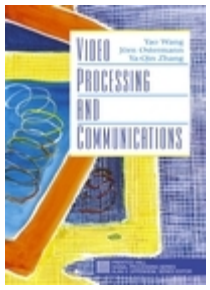
Drift Problem in Scalable Codecs

- Suppose a scalable stream is generated where each new frame is predicted from the complete bit stream of the previous frame and the error is coded into scalable streams.
- For a previous frame, a user receives only partial bit stream
- Following frames will suffer from reference mismatch or drift
- How to solve the problem?

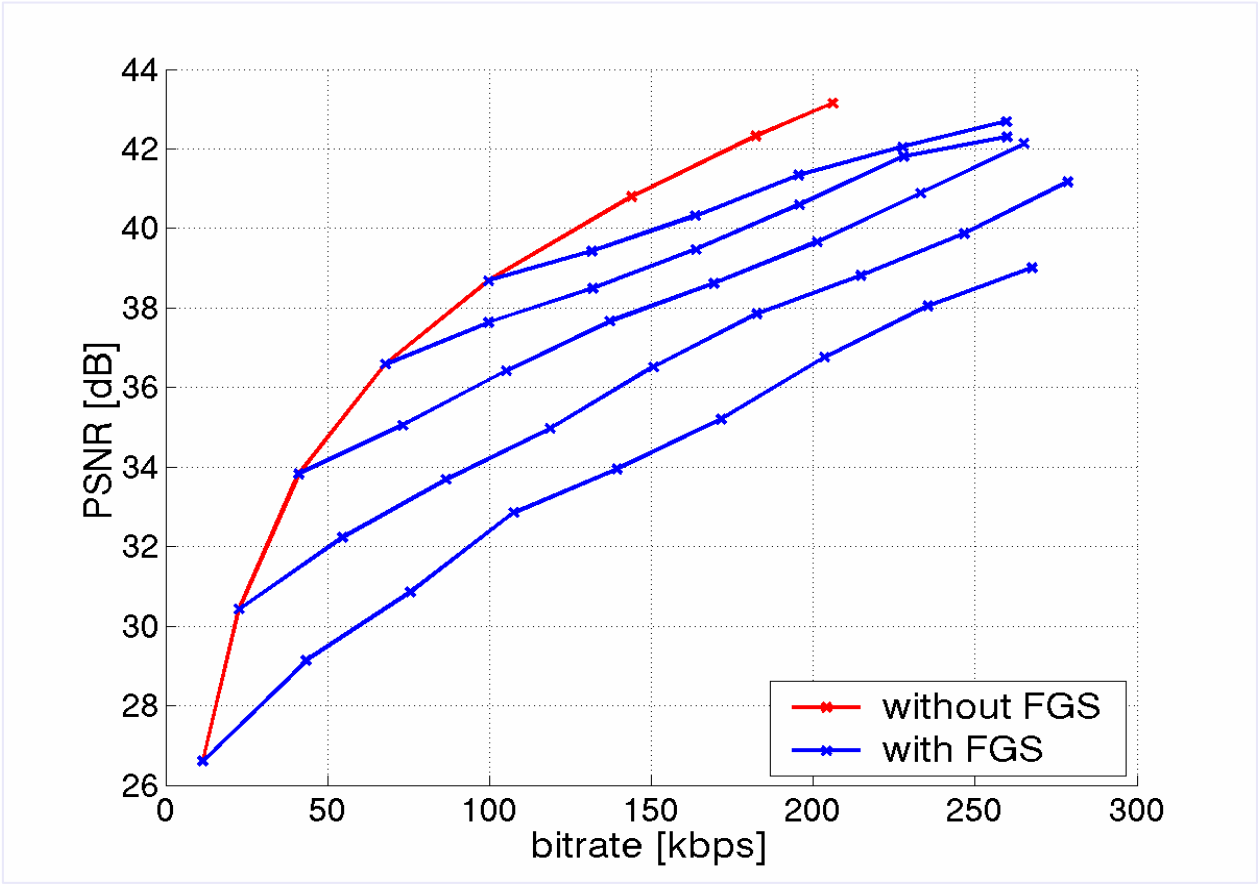


How to Solve the Drift Problem?

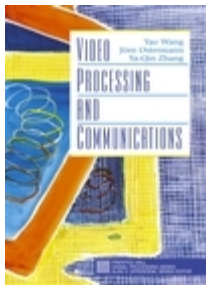
- Predict from only a small part of the previous frame (called base-layer, which is assumed always received)
  - Suppress drift
  - Low prediction accuracy -> low coding efficiency!
  - This is the approach of MPEG4 FGS
    - Coding efficiency substantially lower than a single layer MPEG4 if the base layer rate is low
- Challenge:
  - Provide the best trade-off between drift and coding efficiency.



Trade off Between Coding Efficiency and Drift

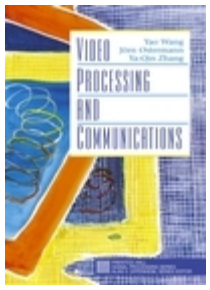


Each blue curve is obtained with MPEG4 FGS using different base-layer rate



Scalable Video Coding Using Wavelet Transforms

- Wavelet-based image coding:
  - Full frame image transform (as opposed to block-based transform)
  - Bit plane coding of the transform coefficients can lead to embedded bitstreams
  - EZW -> SPIHT -> JPEG2000
- Wavelet-based video coding
  - Temporal filtering with and without motion compensation
    - Using MC limits the range of scalability
  - Can achieve temporal, spatial, and quality scalability simultaneously
  - Still an active research activity!



Homework

- Reading assignment: Sec. 9.3, 11.1
- Written assignment
  - Prob. 9.13, 11.3, 11.4
- Computer assignment
  - Prob. 9.11, 9.12
  - Optional: 9.15