ECE160

Multimedia

Lecture 10: Spring 2011

Basic Video Compression Techniques

H.261, MPEG-1 and MPEG-2

Introduction to Video Compression

- A video consists of a time-ordered sequence of frames, i.e., images.
- An obvious solution to video compression would be predictive coding based on previous frames.
- Compression proceeds by subtracting images: subtract in time order and code the residual error.
- It can be done even better by searching for just the right parts of the image to subtract from the previous frame.

Video Compression with Motion Compensation

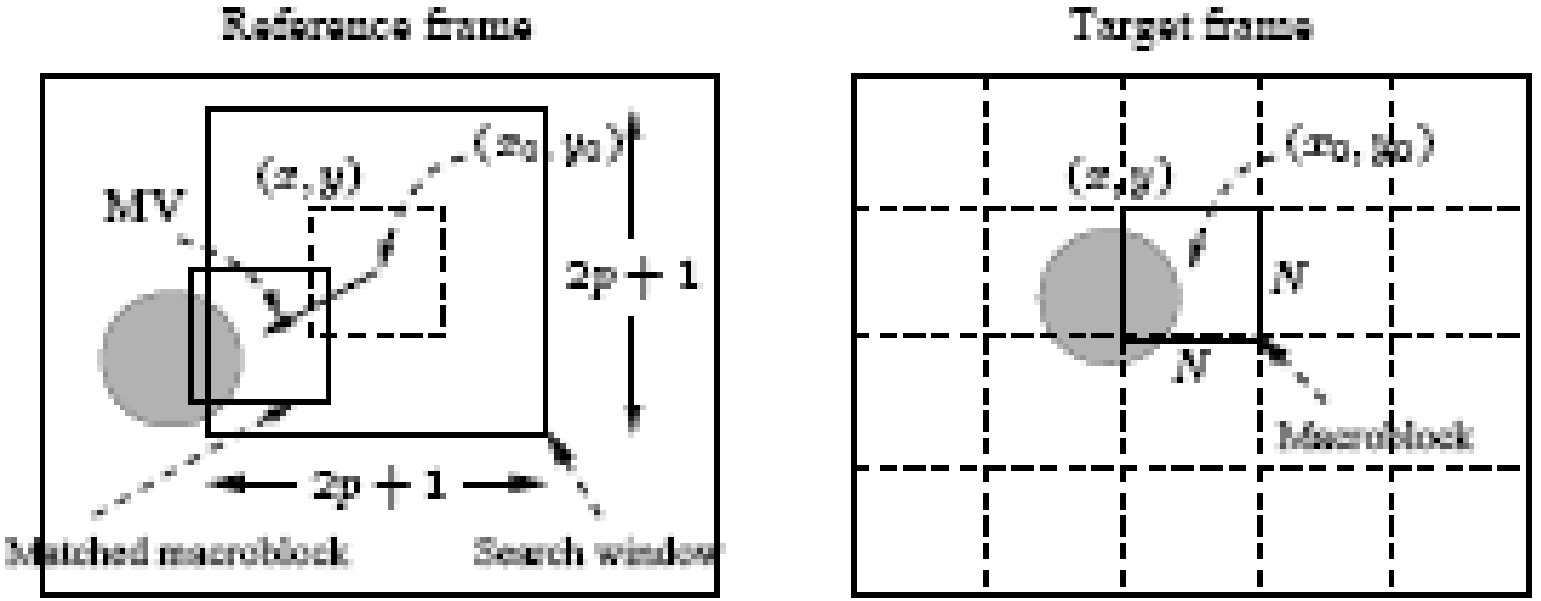
- Consecutive frames in a video are similar - temporal redundancy exists.
- **Temporal redundancy** is exploited so that not every frame of the video needs to be coded independently as a new image.
- The difference between the current frame and other frame(s) in the sequence will be coded - small values and low entropy, good for compression.
- Steps of Video compression based on *Motion Compensation (MC)*:
  1. Motion Estimation (motion vector search).
  2. MC-based Prediction.
  3. Derivation of the prediction error, i.e., the difference.

Motion Compensation

- Each image is divided into *macroblocks* of size *NxN*.
  - By default, *N* = 16 for luminance images. For chrominance images, *N* = 8 if 4:2:0 chroma subsampling is adopted.
- Motion compensation operates at the macroblock level.
  - The current image frame is referred to as *Target Frame*.
  - A match is sought between the macroblock in the Target Frame and the most similar macroblock in previous and/or future frame(s) (referred to as *Reference frame(s)*).
  - The displacement of the reference macroblock to the target macroblock is called a *motion vector* **MV**.

Motion Compensation

- Macroblocks and Motion Vector in Video Compression.



- MV search is usually limited to a small immediate neighborhood – both horizontal and vertical displacements in the range [−*p*,*p*]. This makes a search window of size (2*p*+1)(2*p*+1).

Search for Motion Vectors

- The difference between two macroblocks can then be measured by their *Mean Absolute Difference (MAD)*:

$$MAD(i,j) = \frac{1}{N^2} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} |C(x+k, y+l) - R(x+i+k, y+j+l)|$$

N – size of the macroblock,

k and l – indices for pixels in the macroblock,

i and j – horizontal and vertical displacements,

$C(x+k, y+l)$ – pixels in macroblock in Target frame,

$R(x+i+k, y+j+l)$ – pixels in macroblock in Reference frame

- The goal of the search is to find a vector (*i,j*) as the motion vector **MV** = (**u,v**), such that *MAD*(*i,j*) is minimum:

$$(u,v) = [\ (i,j)\ |\ MAD(i,j) \ \text{ is minimum}, \ i \in [-p,p], \ j \in [-p,p]\ ]$$

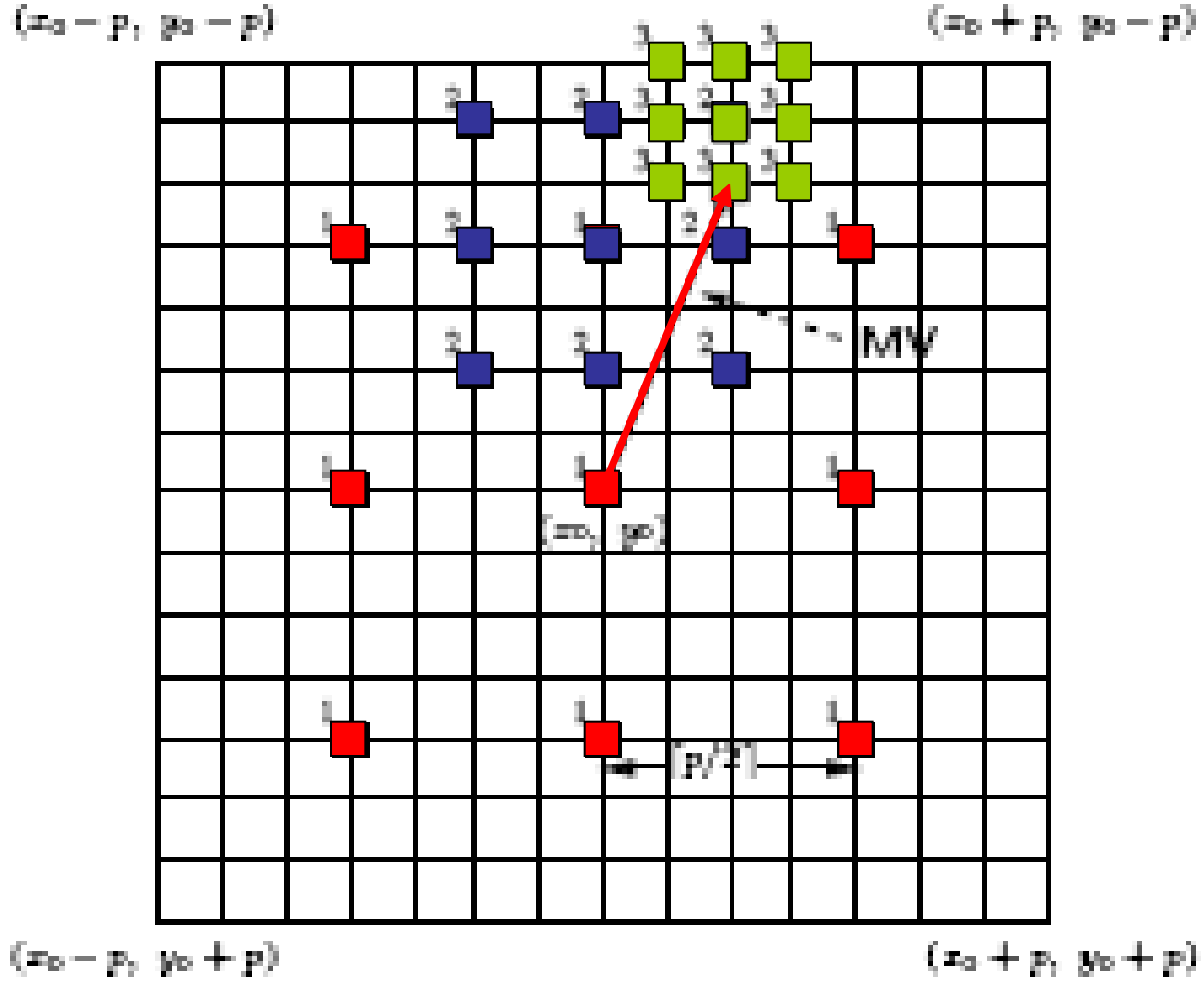
Sequential Search

- **Sequential search**: sequentially search the whole (2*p*+1)x(2*p*+1) window in the Reference frame (also referred to as Full search).
- A macroblock centered at each of the positions within the window is compared to the macroblock in the Target frame pixel by pixel and their respective *MAD* is then derived using the equation above.
- The vector (*i,j*) that offers the least *MAD* is designated as the **MV** (*u,v*) for the macroblock in the Target frame.
- Sequential search method is very costly - assuming each pixel comparison requires three operations (subtraction, absolute value, addition), the cost for obtaining a motion vector for a single macroblock is $(2p+1).(2p+1).N^2.3$) => $O(p^2N^2)$.

Logarithmic Search

- **Logarithmic search**: a cheaper version, that is suboptimal but still usually effective.
- The procedure for 2D Logarithmic Search of motion vectors takes several iterations and is akin to a binary search:
  - As illustrated in the Figure below, initially only nine locations in the search window are used as seeds for a MAD-based search; they are marked as `1'.
  - After the one that yields the minimum *MAD* is located, the center of the new search region is moved to it and the step-size ("offset") is reduced to half.
  - In the next iteration, the nine new locations are marked as `2', and so on.

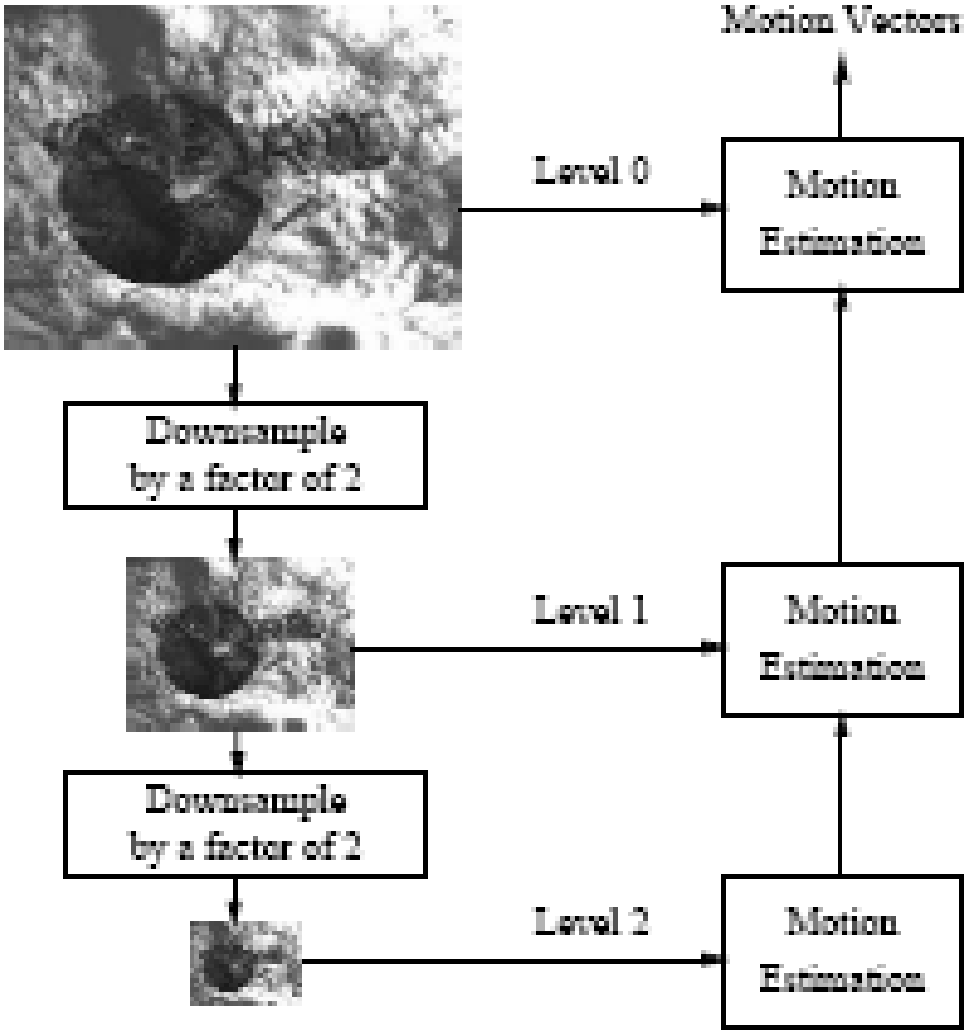
2D Logarithmic Search for Motion Vectors



Hierarchical Search

- The search can benefit from a hierarchical (multiresolution) approach in which initial estimation of the motion vector is obtained from images with a significantly reduced resolution.
- The Figure below shows a three-level hierarchical search in which the original image is at Level 0, images at Levels 1 and 2 are obtained by down-sampling from the previous levels by a factor of 2, and the initial search is conducted at Level 2.
- Since the size of the macroblock is smaller and *p* can also be proportionally reduced, the number of operations required is greatly reduced.

Three-level Hierarchical Search for Motion Vectors



Comparison of Computational Cost of Motion Vector Search

Search Method	OPS_per_second for 720×480 at 30 fps	
	$p = 15$	$p = 7$
Sequential search	29.89×10^9	7.00×10^9
2D Logarithmic search	1.25×10^9	0.78×10^9
3-level Hierarchical search	0.51×10^9	0.40×10^9

H.261

H.261: An earlier digital video compression standard, its principle of MC-based compression is retained in all later video compression standards.

- The standard was designed for videophone, video conferencing and other audiovisual services over ISDN.
- The video codec supports bit-rates of *px*64 kbps, where *p* ranges from 1 to 30 (Hence also known as *px*64).
- Require that the delay of the video encoder be less than 150 msec so that the video can be used for real-time bidirectional video conferencing.

H.261 Video Formats

H.261 belongs to the following set of ITU recommendations for visual telephony systems:

1. H.221 - Frame structure for an audiovisual channel supporting 64 to 1,920 kbps.
2. H.230 - Frame control signals for audiovisual systems.
3. H.242 - Audiovisual communication protocols.
4. H.261 - Video encoder/decoder for audiovisual services at *px*64 kbps.
5. H.263 - Improved video coding standard for video conferencing at bit-rates of less than 64 kbps.
6. H.320 - Narrow-band audiovisual terminal equipment for *px*64 kbps transmission.

Video Formats Supported by H.261

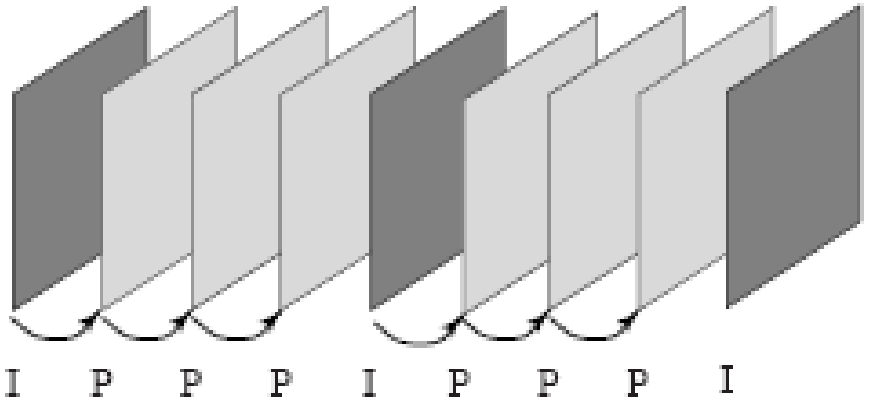
Video format	Luminance image resolution	Chrominance image resolution	Bit-rate (Mbps) (if 30 fps and uncompressed)	H.261 support
QCIF	176×144	88×72	9.1	required
CIF	352×288	176×144	36.5	optional

H.261 Frame Sequence

Two types of image frames are defined: Intra-frames (**I-frames**) and Inter-frames (**P-frames**):

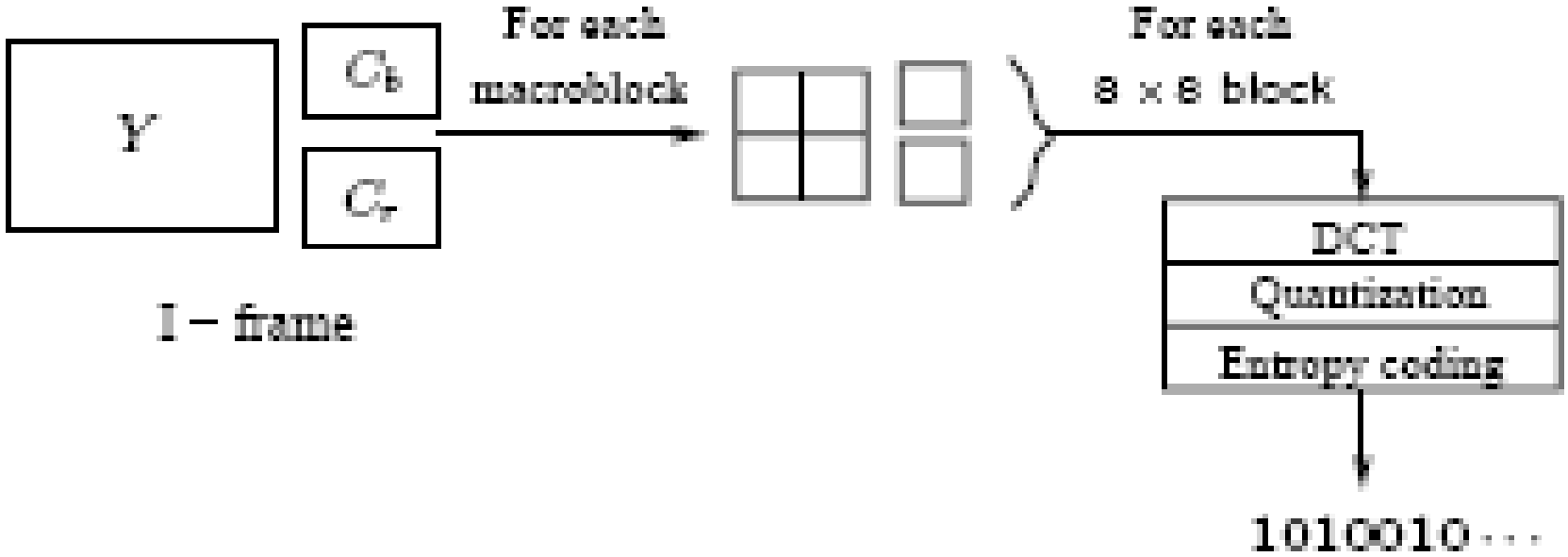
- I-frames are treated as independent images. Transform coding method similar to JPEG is applied within each I-frame, hence “Intra".
- P-frames are not independent: coded by a forward predictive coding method (prediction from a previous P-frame is allowed - not just from a previous I-frame).
- **Temporal redundancy removal** is included in P-frame coding, whereas I-frame coding performs only **spatial redundancy removal**.
- To avoid propagation of coding errors, an I-frame is usually sent a couple of times in each second of the video.

- Motion vectors in H.261 are always measured in units of full pixel and they have a limited range of 15 pixels, i.e., $p = 15$.



Intra-frame (I-frame) Coding

- **Macroblocks** are of size 16x16 pixels for the Y frame, and 8x8 for Cb and Cr frames, since 4:2:0 chroma subsampling is employed. A macroblock consists of four Y, one Cb, and one Cr 8x8 blocks.
- For each 8x8 block a DCT transform is applied, the DCT coefficients then go through quantization zigzag scan

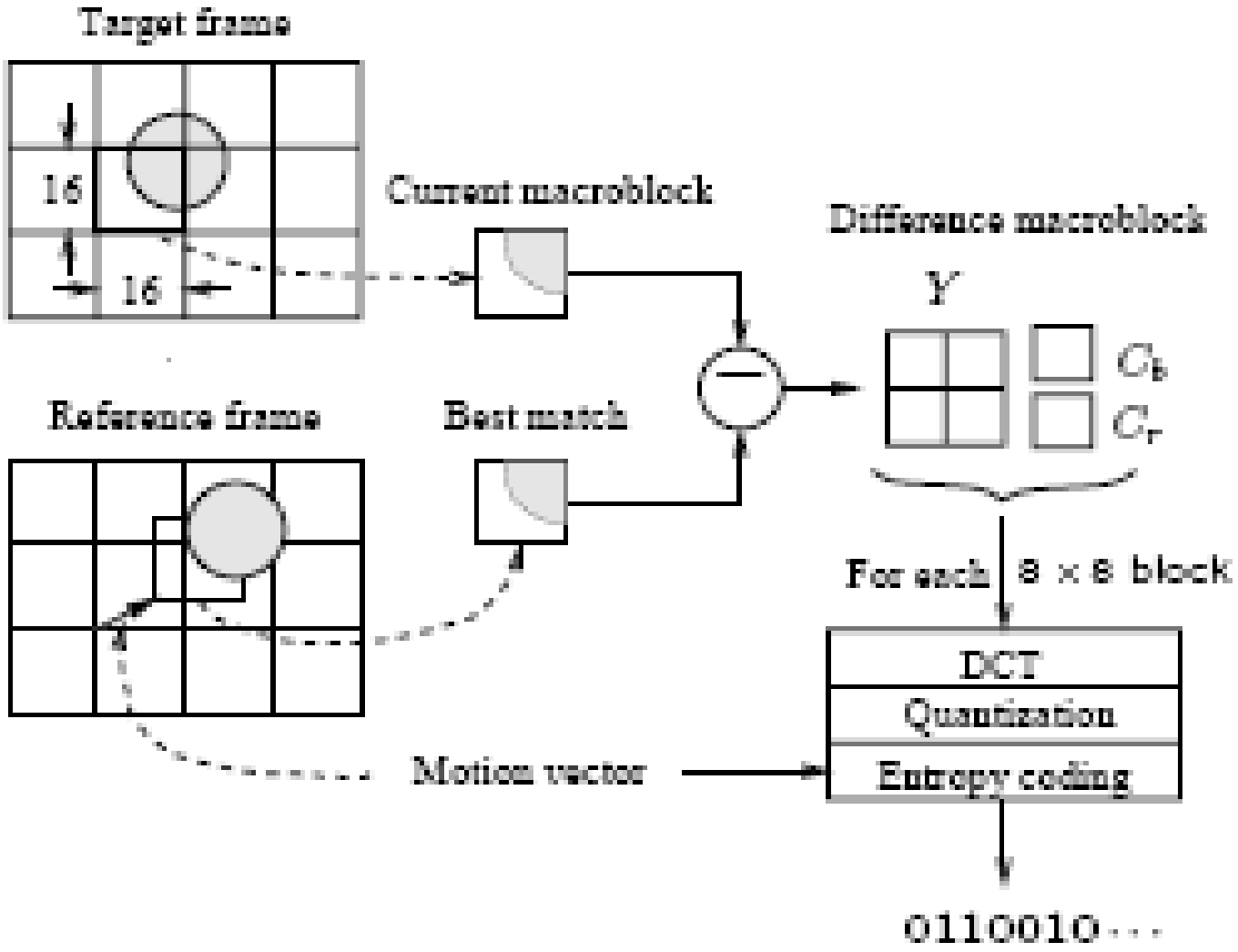


Inter-frame (P-frame) Predictive Coding

- For each macroblock in the Target frame, a motion vector is found by one of the search methods discussed earlier.
- After the prediction, a *difference macroblock* is derived to measure the *prediction error*.
- Each of these 8x8 blocks go through DCT, quantization, zigzag scan and entropy coding procedures.
- The P-frame coding encodes the difference macroblock (not the Target macroblock itself).
- Sometimes, a good match cannot be found, i.e., the prediction error exceeds an acceptable level.
  - The MB itself is then encoded (treated as an Intra MB) and in this case it is termed a *non-motion compensated MB*.
- For motion vector, the difference **MVD** is sent for entropy coding:

$$\mathbf{MVD} = \mathbf{MV}_{\mathbf{Preceding}} - \mathbf{MV}_{\mathbf{Current}}$$

P-frame Coding based on Motion Compensation.



H.263

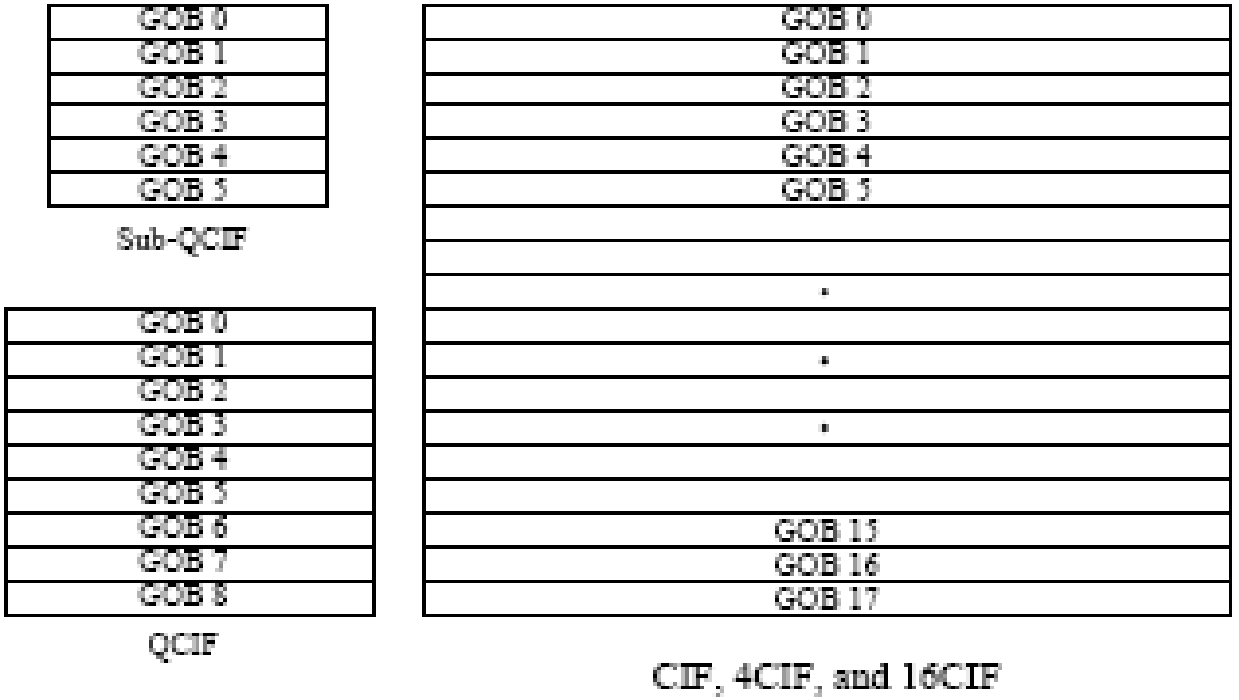
- H.263 is an improved video coding standard for video conferencing and other audiovisual services transmitted on Public Switched Telephone Networks (PSTN).
  - Aims at low bit-rate communications at bit-rates of less than 64 kbps.
  - Uses predictive coding for inter-frames to reduce temporal redundancy and transform coding for the remaining signal to reduce spatial redundancy (for both Intra-frames and inter-frame prediction).

Video Formats supported by H.263

Video format	Luminance image resolution	Chrominance image resolution	Bit-rate (Mbps) (if 30 fps and uncompressed)	Bit-rate (kbps) BPPmaxKb (compressed)
sub-QCIF	128×96	64×48	4.4	64
QCIF	176×144	88×72	9.1	64
CIF	352×288	176×144	36.5	256
4CIF	704×576	352×288	146.0	512
16CIF	$1,408 \times 1,152$	704×576	583.9	1024

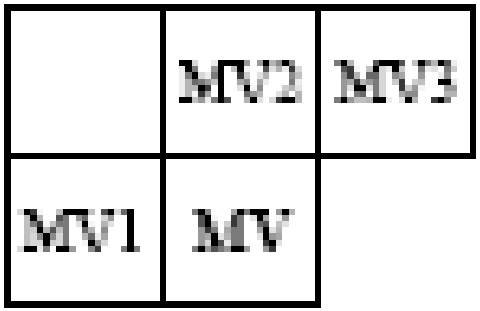
H.263 & Group of Blocks (GOB)

- Like H.261, H.263 standard also supports Group of Blocks (GOB).
- The difference is that GOBs in H.263 do not have a fixed size, and they always start and end at the left and right borders of the picture.
- Each QCIF luminance image consists of 9 GOBs and each GOB has 111 MBs (17616 pixels), whereas each 4CIF luminance image consists of 18 GOBs and each GOB has 44x2 MBs (704x32 pixels).



Motion Compensation in H.263

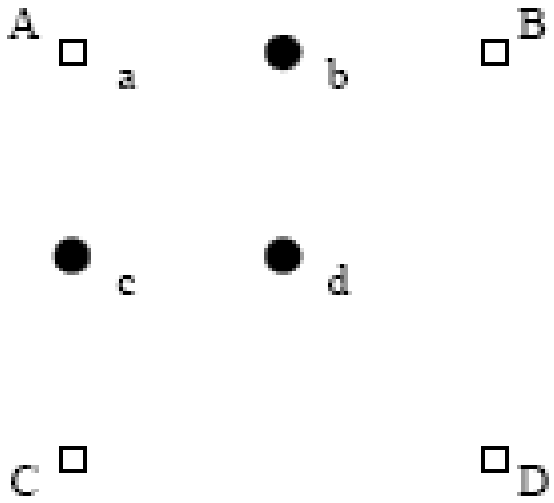
- The horizontal and vertical components of the **MV** are predicted from the median values of the horizontal and vertical components, respectively, of **MV1**, **MV2**, **MV3** from the “previous", “above" and “above and right" MBs.
- For the Macroblock with **MV**(u,v):$u_p = median(u_1,u_2,u_3)$, $v_p = median(v_1,v_2,v_3)$.
- Instead of coding the **MV**(u,v) itself, the error vector (u,v) is coded, where $u = u - u_p$ and $v = v - v_p$.



MV	Current motion vector
MV1	Previous motion vector
MV2	Above motion vector
MV3	Above and right motion vector

Half-Pixel Precision

- In order to reduce the prediction error, **half-pixel precision** is supported in H.263 vs. full-pixel precision only in H.261.
  - The default range for both the horizontal and vertical components *u* and *v* of **MV**(*u,v*) are now [−16,15.5].
  - The pixel values needed at half-pixel positions are generated by a simple **bilinear interpolation** method.



Full-pixel position

Half-pixel position

$a = A$

$b = (A + B + 1) / 2$

$c = (A + C + 1) / 2$

$d = (A + B + C + D + 2) / 4$

MPEG Video Coding I
MPEG-1 and 2

- **MPEG**: *Moving Pictures Experts Group*, established in 1988 for the development of digital video.
- It is appropriately recognized that proprietary interests need to be maintained within the family of MPEG standards:
- Accomplished by defining only a compressed bitstream that implicitly defines the decoder.
- The compression algorithms, and thus the encoders, are completely up to the manufacturers.

MPEG-1

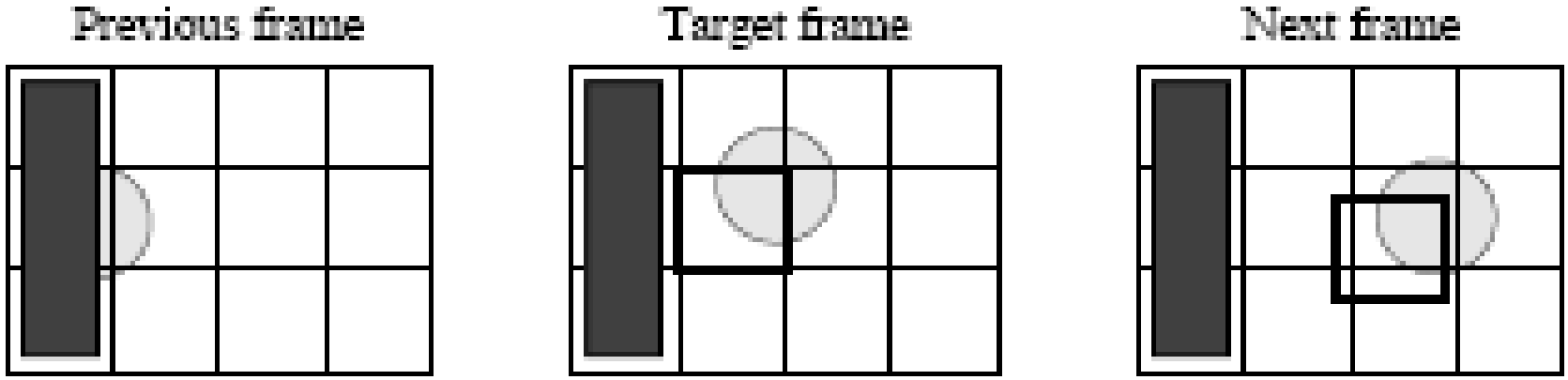
- MPEG-1 adopts the CCIR601 digital TV format also known as SIF (*Source Input Format*).
- MPEG-1 supports only non-interlaced video. Normally, its picture resolution is:
  - 352x240 for NTSC video at 30 fps
  - 352x288 for PAL video at 25 fps
  - It uses 4:2:0 chroma subsampling
- The MPEG-1 standard is also ISO/IEC 11172. It has five parts:
  11172-1 Systems, 11172-2 Video,
  11172-3 Audio, 11172-4 Conformance,
  11172-5 Software.

Motion Compensation in H.261

- Motion Compensation (MC) based video encoding in H.261 works as follows:
- In Motion Estimation (ME), each macroblock (MB) of the Target P-frame is assigned a best matching MB from the previously coded I or P frame - prediction.
- **prediction error:** The difference between the MB and its matching MB, sent to DCT and its subsequent encoding steps.
- The prediction is from a previous frame - **forward prediction**.

Motion Compensation in MPEG-1

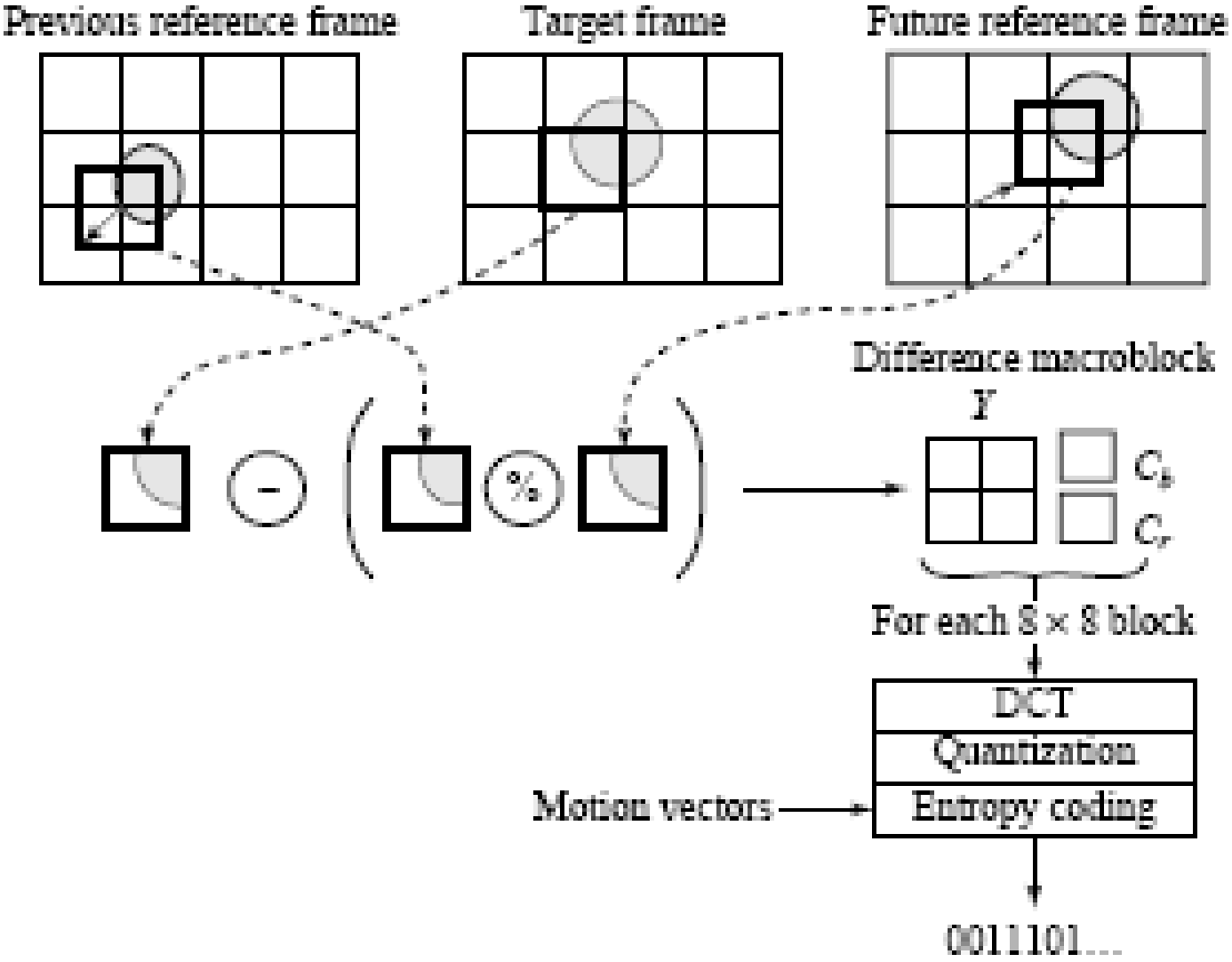
- The Need for Bidirectional Search.
  - The MB containing part of a ball in the Target frame cannot find a good matching MB in the previous frame because half of the ball was occluded by another object. A match however can readily be obtained from the next frame.



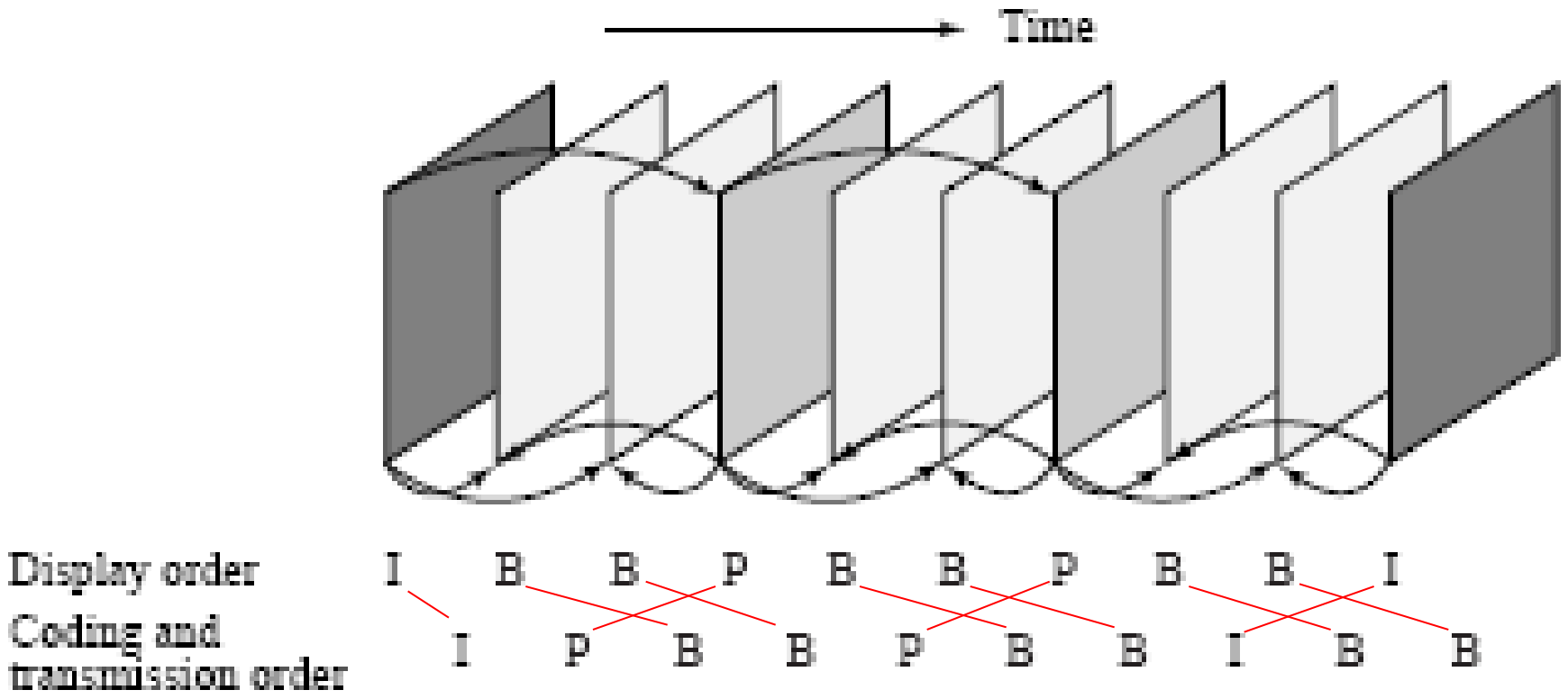
Motion Compensation in MPEG-1

- MPEG introduces a third frame type - *B-frames*, and its accompanying bi-directional motion compensation.
- Each MB from a B-frame will have up to *two* motion vectors (MVs) (one from the forward and one from the backward prediction).
- If matching in both directions is successful, then two MVs will be sent and the two corresponding matching MBs are averaged before comparing to the Target MB for generating the prediction error.
- If an acceptable match can be found in only one of the reference frames, then only one MV and its corresponding MB will be used from either the forward or backward prediction.

Motion Compensation in MPEG-1



MPEG Frame Sequence.



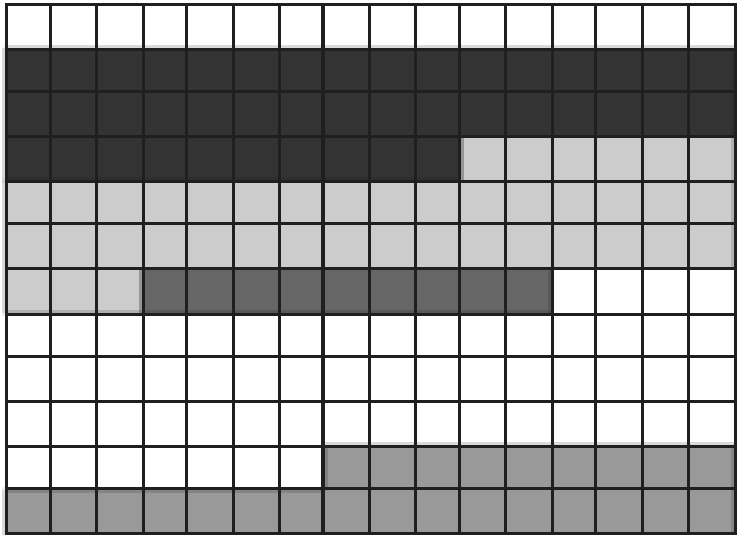
Major Differences from H.261

- Source formats supported:
- H.261 only supports CIF (352x288) and QCIF (176x144) source formats, MPEG-1 supports SIF (352x240 for NTSC, 352x288 for PAL).
- MPEG-1 also allows specification of other formats as long as the Constrained Parameter Set (CPS) as shown below is satisfied:

Parameter	Value
Horizontal size of picture	≤ 768
Vertical size of picture	≤ 576
No. of MBs / picture	≤ 396
No. of MBs / second	$\leq 9,900$
Frame rate	≤ 30 fps
Bit-rate	$\leq 1,856$ kbps

Major Differences from H.261

- Instead of GOBs as in H.261, an MPEG-1 picture can be divided into one or more **slices**:
  - May contain variable numbers of macroblocks in a single picture.
  - May also start and end anywhere as long as they fill the whole picture.
  - Each slice is coded independently - additional flexibility in bit-rate control.
  - Slice concept is important for error recovery.



Major Differences from H.261

- Quantization:

  - MPEG-1 quantization uses different quantization tables for its Intra and Inter coding

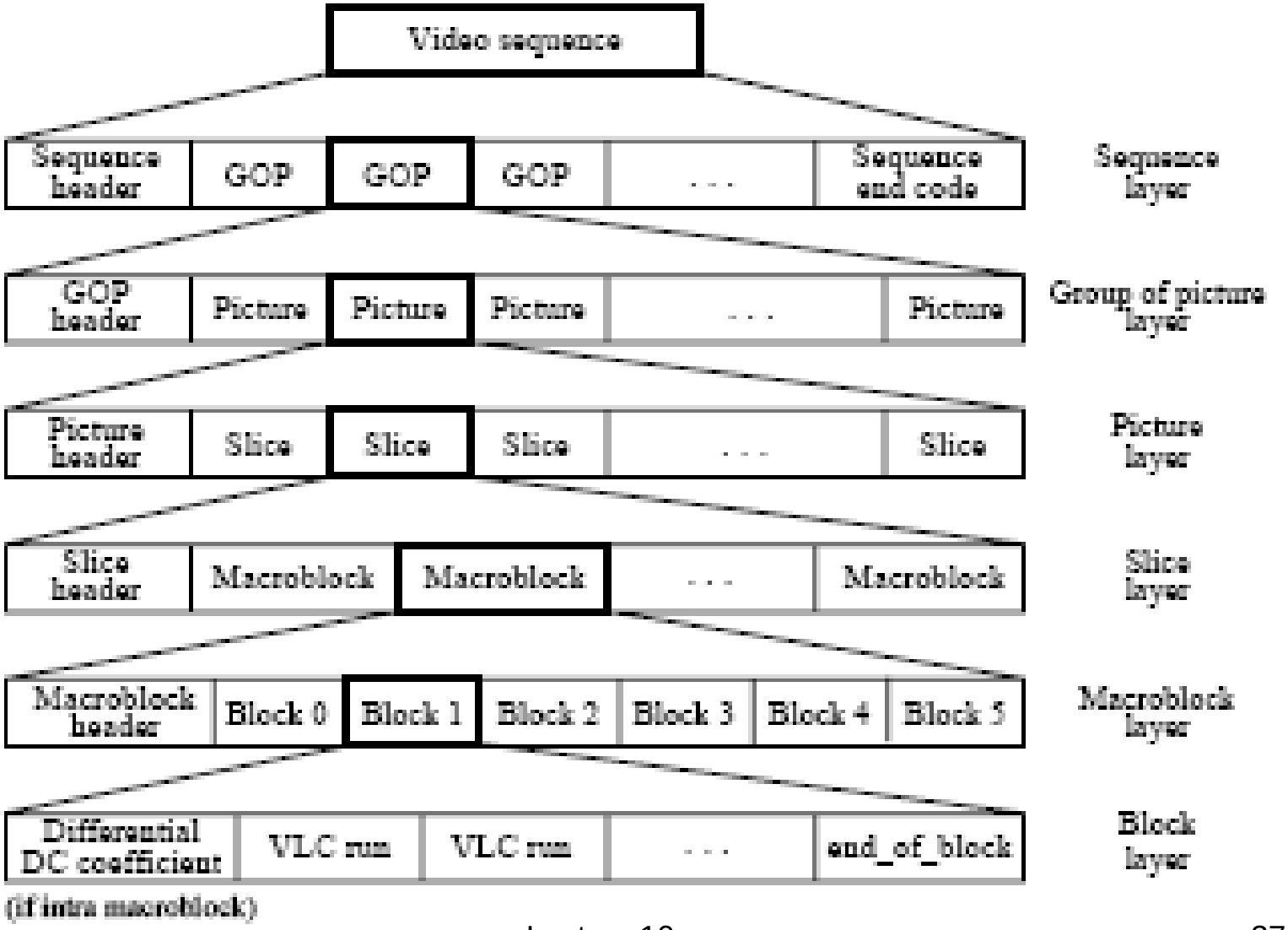
- MPEG-1 allows motion vectors to be sub-pixel precision (1/2 pixel). The technique of “bilinear interpolation" (H.263) is used to generate the values at half-pixel locations.
- Compared to the maximum of 15 pixels for motion vectors in H.261, MPEG-1 supports a range of
  [−512, 511.5] for half-pixel precision and
  [−1024, 1023] for full-pixel precision motion vectors.
- The MPEG-1 bitstream allows random access –
  In the GOP layer, each GOP is time coded.

Typical Sizes of MPEG-1 Frames

- The typical size of compressed P-frames is significantly smaller than that of I-frames - because temporal redundancy is exploited in inter-frame compression.
- B-frames are even smaller than P-frames - because
  (a) the advantage of bidirectional prediction and
  (b) the lowest priority given to B-frames.

Type	Size	Compression
I	18 kB	7:1
P	6 kB	20:1
B	2.5 kB	50:1
Avg	4.8 kB	27:1

Layers of MPEG-1 Video Bitstream



MPEG-2 Profiles

- **MPEG-2**: For higher quality video at a bit-rate of more than 4 Mbps.
  - Defined seven **profiles** aimed at different applications:
  - **Simple, Main, SNR scalable, Spatially scalable, High, 4:2:2, Multiview**.
  - Within each profile, up to four *levels* are defined
  - The DVD video specification allows only four display resolutions: 720x480, 704x480, 352x480, and 352x240 - a restricted form of the MPEG-2 Main profile at the Main and Low levels.

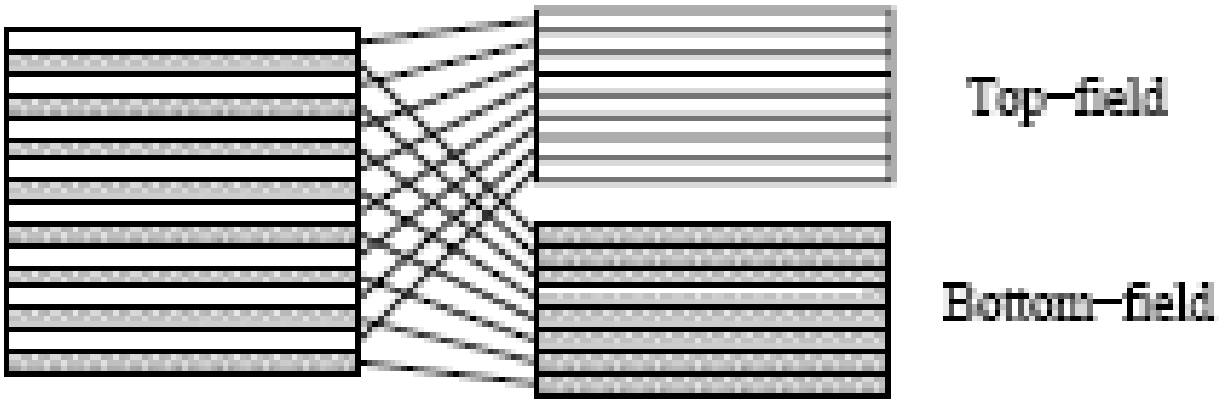
Profiles and Levels in MPEG-2

Level	Simple Profile	Main Profile	SNR Scalable Profile	Spatially Scalable Profile	High Profile	4:2:2 Profile	Multiview Profile
High		*			*		
High 1440		*		*	*		
Main	*	*	*		*	*	*
Low		*	*				

Level	Max Resolution	Max fps	Max Pixels/sec	Max coded Data Rate (Mbps)	Application
High	$1,920 \times 1,152$	60	62.7×10^6	80	film production
High 1440	$1,440 \times 1,152$	60	47.0×10^6	60	consumer HDTV
Main	720×576	30	10.4×10^6	15	studio TV
Low	352×288	30	3.0×10^6	4	consumer tape equiv.

Supporting Interlaced Video

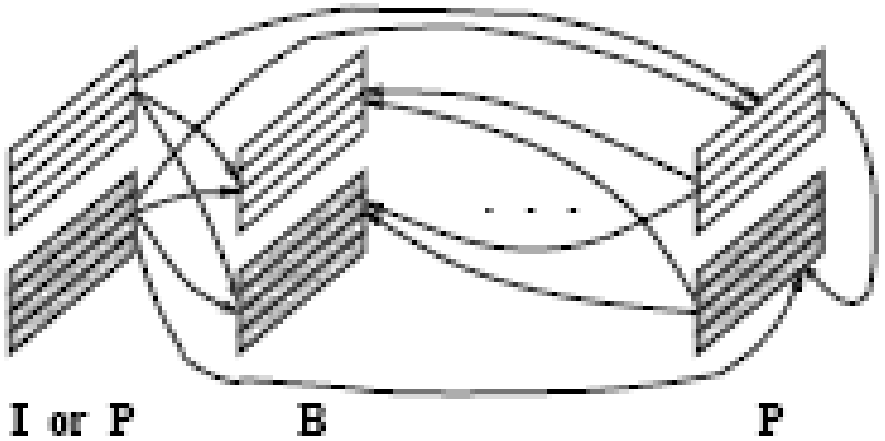
- MPEG-2 must support interlaced video as well since this is one of the options for digital broadcast TV and HDTV.
- In interlaced video each frame consists of two fields, referred to as the *top-field* and the *bottom-field*.
  - In a *Frame-picture*, all scanlines from both fields are interleaved to form a single frame, then divided into 16x16 macroblocks and coded using MC.
  - If each field is treated as a separate picture, then it is called *Field-picture*.



Five Modes of Prediction

- MPEG-2 defines **Frame Prediction** and **Field Prediction** as well as five prediction modes:

1. **Frame Prediction for Frame-pictures:** Identical to MPEG-1 MC-based prediction methods in both P-frames and B-frames.
2. **Field Prediction for Field-pictures:** A macroblock size of 16x16 from Field-pictures is used.



Five Modes of Prediction

3. **Field Prediction for Frame-pictures:** The top-field and bottom-field of a Frame-picture are treated separately. Each 16x16 macroblock (MB) from the target Frame-picture is split into two 16x8 parts, each coming from one field. Field prediction is carried out for these 16x8 parts.
4. **16x8 MC for Field-pictures:** Each 16x16 macroblock (MB) from the target Field-picture is split into top and bottom 16x8 halves. Field prediction is performed on each half. This generates two motion vectors for each 16x16 MB in the P-Field-picture, and up to four motion vectors for each MB in the B-Field-picture.

   This mode is good for a finer MC when motion is rapid and irregular.

Five Modes of Prediction

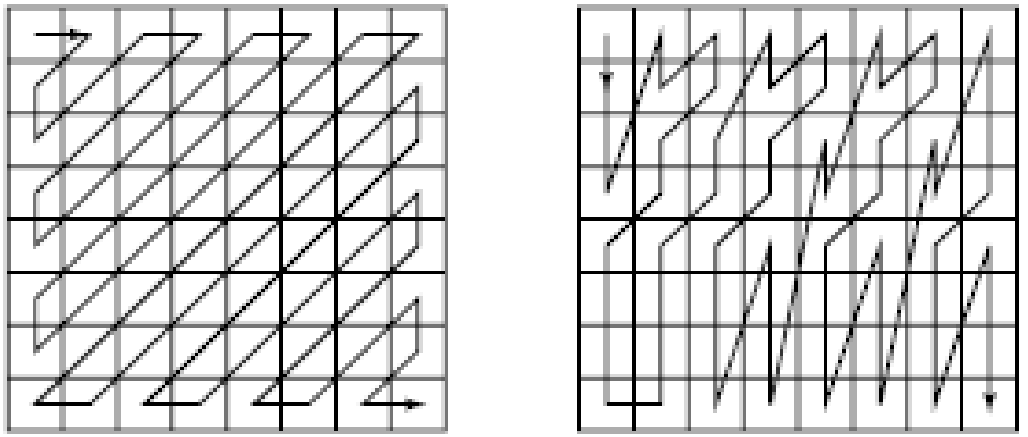
5. **Dual-Prime for P-pictures:** First, Field prediction is made from each previous field with the same parity (top or bottom). Each motion vector **mv** is then used to derive a calculated motion vector **cv** in the field with the opposite parity taking into account the temporal scaling and vertical shift between lines in the top and bottom fields.

   For each MB, the pair **mv** and **cv** yields two preliminary predictions. Their prediction errors are averaged and used as the final prediction error.

- This mode mimics B-picture prediction for P-pictures without adopting backward prediction (and hence with less encoding delay).

Alternate Scan and Field DCT

- Techniques aimed at improving the effectiveness of DCT on prediction errors, only applicable to Frame-pictures in interlaced videos:
  - Due to the nature of interlaced video the consecutive rows in the 8x8 blocks are from different fields, there exists less correlation between them than between the alternate rows.
  - Alternate scan recognizes the fact that in interlaced video the vertically higher spatial frequency components may have larger magnitudes and thus allows them to be scanned earlier in the sequence.
- In MPEG-2, **Field DCT** can also be used to address the same issue.



MPEG-2 Scalabilities

- The MPEG-2 **scalable coding**: A base layer and one or more enhancement layers can be defined - also known as **layered coding**.
  - The base layer can be independently encoded, transmitted and decoded to obtain basic video quality.
  - The encoding and decoding of the enhancement layer is dependent on the base layer or the previous enhancement layer.
- Scalable coding is especially useful for MPEG-2 video transmitted over networks with following characteristics:
  - Networks with very different bit-rates.
  - Networks with variable bit rate (VBR) channels.
  - Networks with noisy connections.

MPEG-2 Scalabilities

- MPEG-2 supports the following scalabilities:
  1. SNR Scalability - enhancement layer provides higher SNR.
  2. Spatial Scalability - enhancement layer provides higher spatial resolution.
  3. Temporal Scalability - enhancement layer facilitates higher frame rate.
  4. Hybrid Scalability - combination of any two of the above three scalabilities.
  5. Data Partitioning - quantized DCT coefficients are split into partitions.

SNR Scalability

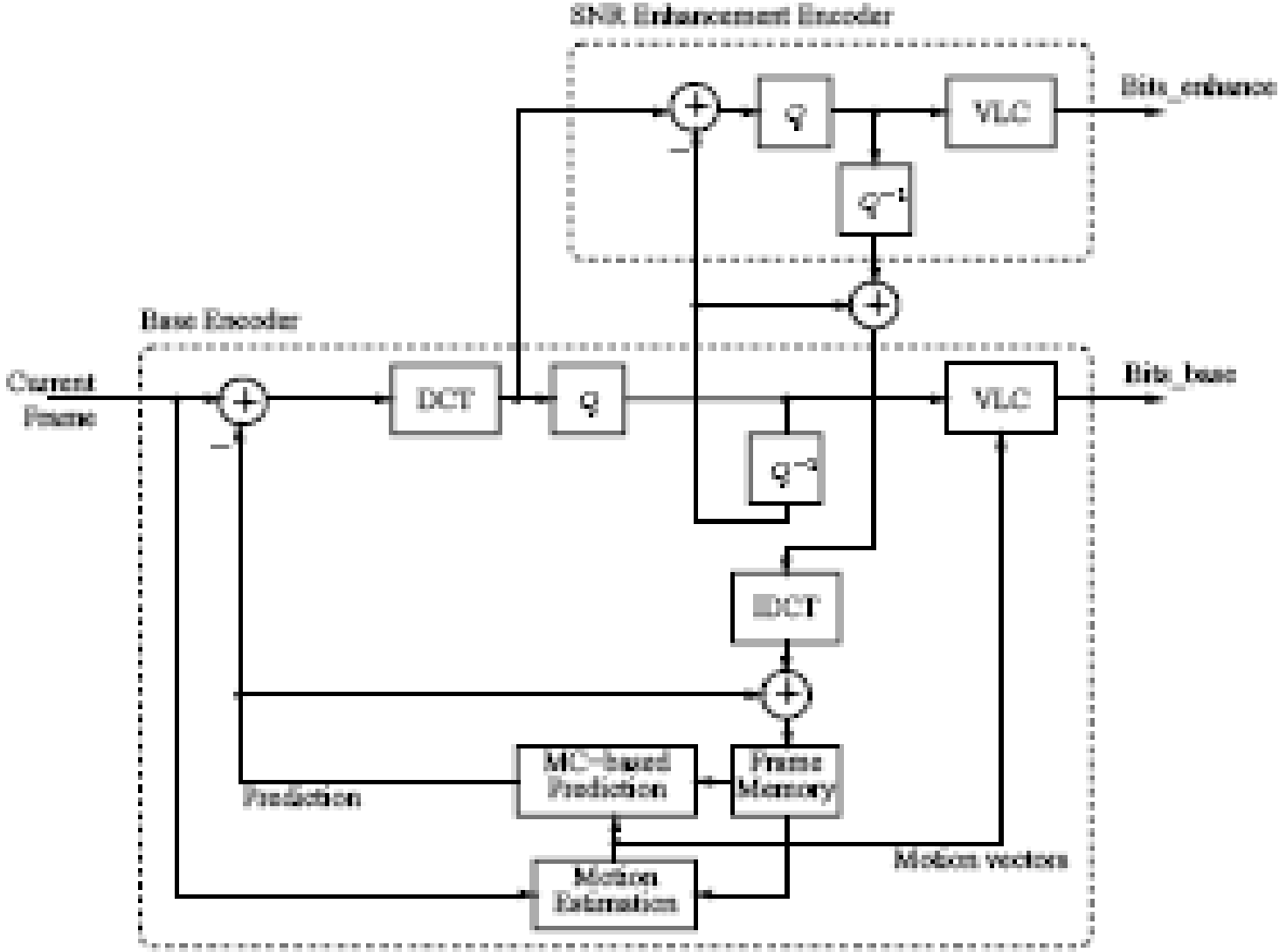
- **SNR scalability**: Refers to the enhancement/refinement over the base layer to improve the Signal-Noise-Ratio (SNR).
- The MPEG-2 SNR scalable encoder will generate output bit-streams *Bits base* and *Bits enhance* at two layers:

  1. At the Base Layer, a coarse quantization of the DCT coefficients is employed which results in fewer bits and a relatively low quality video.

  2. The coarsely quantized DCT coefficients are then inversely quantized (Q^{-1}) and fed to the Enhancement Layer to be compared with the original DCT coefficient.

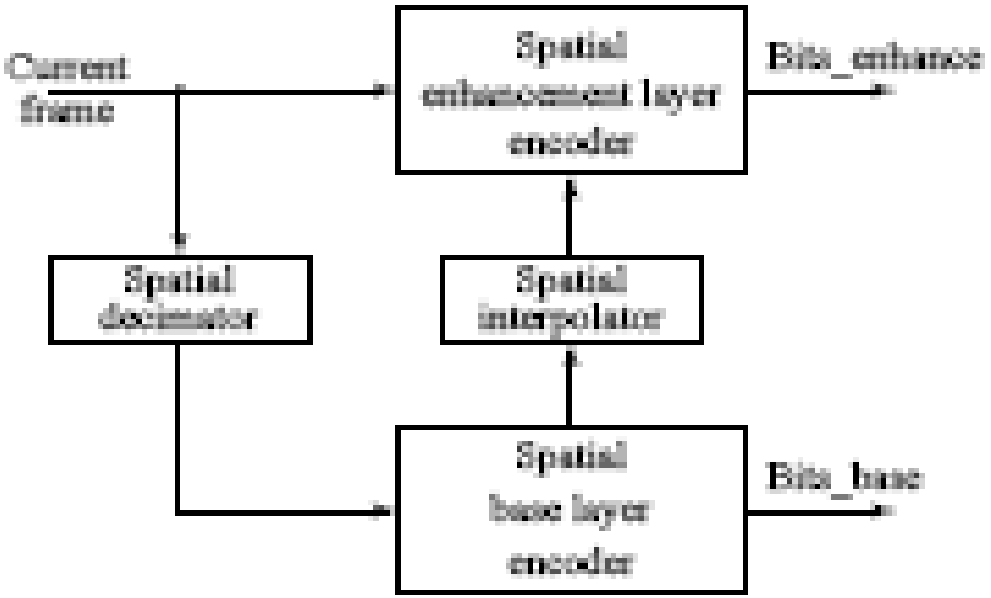
  3. Their difference is finely quantized to generate a **DCT coefficient refinement**, which, after VLC, becomes the bitstream called Bits_enhance.

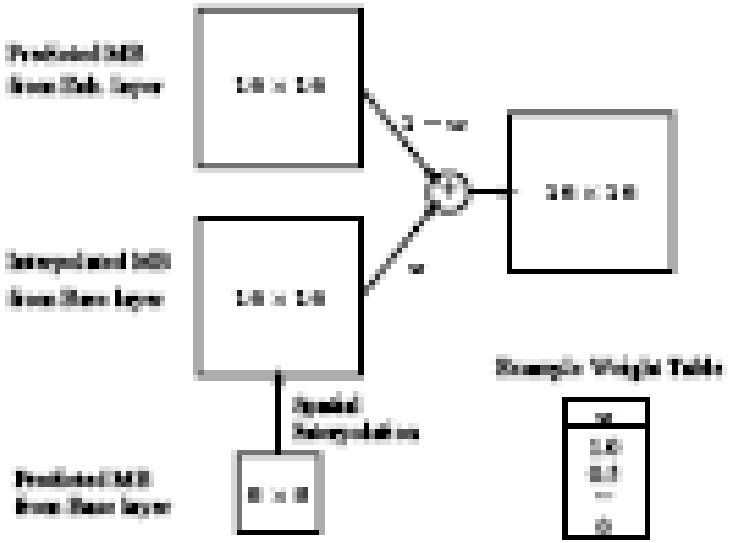
MPEG-2 SNR Scalability (Encoder)



Spatial Scalability

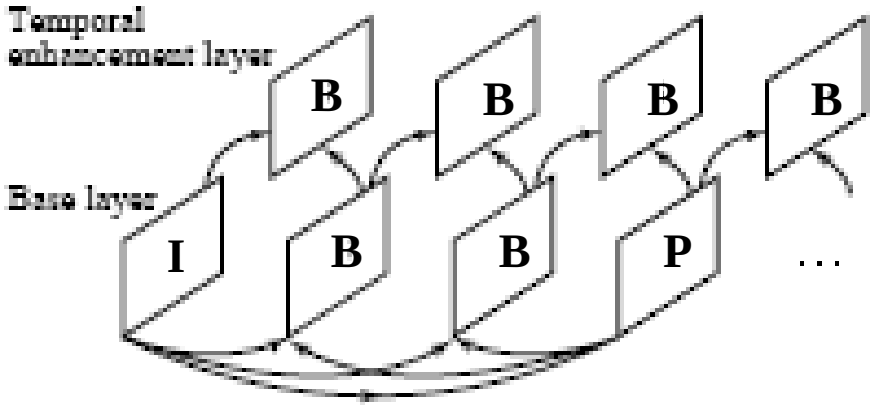
- The base layer is generates a bitstream of reduced-resolution pictures. When combined by the enhancement layer, pictures at the original resolution are produced.
- The Base and Enhancement layers for MPEG-2 spatial scalability are not tightly coupled as in SNR scalability.

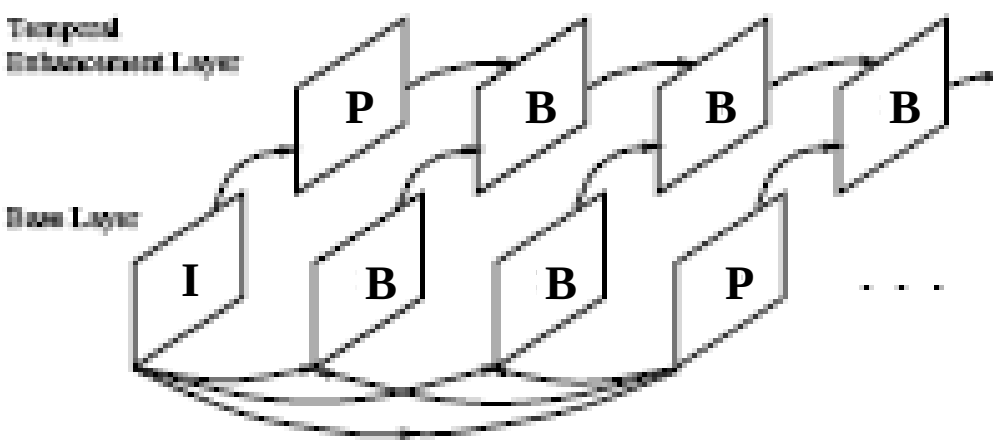




Temporal Scalability

- The input video is temporally demultiplexed into two pieces, each carrying half of the original frame rate.
- Base Layer Encoder carries out the normal single-layer coding procedures for its own input video and yields the output bitstream Bits base.
- The prediction of matching MBs at the Enhancement Layer can be obtained in two ways:
  - Interlayer MC (Motion-Compensated) Prediction
  - Combined MC Prediction and Interlayer MC Prediction





Data Partitioning

- *Base partition* contains lower-frequency DCT coefficients,
- *Enhancement partition* contains high-frequency DCT coefficients.
- Strictly speaking, data partitioning is not layered coding, since a single stream of video data is simply divided up and there is no further dependence on the base partition in generating the enhancement partition.
- Useful for transmission over noisy channels and for progressive transmission.

Other Differences from MPEG-1

- **Better resilience to bit-errors:** In addition to *Program Stream*, a *Transport Stream* is added to MPEG-2 bit streams.
- **Support of 4:2:2 and 4:4:4 chroma subsampling**.
- **More restricted slice structure:** MPEG-2 slices must start and end in the same macroblock row. In other words, the left edge of a picture always starts a new slice and the longest slice in MPEG-2 can have only one row of macroblocks.
- **More flexible video formats:** It supports various picture resolutions as defined by DVD, ATV and HDTV.

Other Differences from MPEG-1

- **Nonlinear quantization -** two types of scales are allowed:

  1. For the first type, *scale* is the same as in MPEG-1 in which it is an integer in the range of [1, 31] and $scale_i = i$.

  2. For the second type, a nonlinear relationship exists, i.e., $scale_i \neq i$.

i	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
$scale_i$	1	2	3	4	5	6	7	8	10	12	14	16	18	20	22	24
i	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	
$scale_i$	28	32	36	40	44	48	52	56	64	72	80	88	96	104	112	