Chapter 5: Compression (Part 3)

Video

Video compression

- We need a video (pictures and sound) compression standard for:
  - teleconferencing
  - digital TV broadcasting
  - video telephone
  - movies
- Motion JPEG
  - compress each frame individually as a still image using JPEG
  - fails to take into consideration the extensive frame-to-frame redundancy present in all video sequences

H.261

- ITU-T H.261 video codec for audiovisual services approved in 1990
- also called p*64
- applications in videophone and video conferencing over ISDN communication
- bandwidth for transmission at $p \times$ 64kbps
  - 1 B-ISDN channel = 64kbps
  - $p = 1$ or 2: suitable for videophone, desktop face-to-face visual communication
  - $p \geq 6$: ok for videoconferencing

H.261

- H.261 was developed for *real-time* encoding and decoding.
- *symmetric* encoding: compression delay ~ decompression delay

ITU-T Video Format for H.261

- Video images are transmitted in $Y'C_RC_B$ components.

	CIF		QCIF	
	Lines	**Pixels**	**Lines**	**Pixels**
Luminance (Y')	288	352	144	176
Chrominance (C_b)	144	176	72	88
Chrominance (C_r)	144	176	72	88

- Frame rate: CIF: 30fps; QCIF: 15/7.5 fps.
- All H.261 implementations must be able to encode QCIF; CIF (Common Intermediate Format) is optional.

H.261

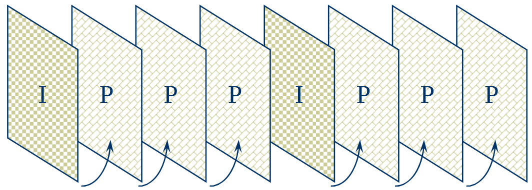
- If uncompressed:
  - CIF at 30fps requires
    - $(288 \times 352 \times 8 + 144 \times 176 \times 8 + 144 \times 176 \times 8) \times 30 \approx 37$ Mbps
  - QCIF at 15fps requires
    - $(144 \times 176 \times 8 + 72 \times 88 \times 8 + 72 \times 88 \times 8) \times 15 \approx 4.7$ Mbps
- ISDN can support 1×64kbps up to 30×64kbps = 2Mbps, therefore the bandwidth is insufficient and compression is required.

Compression requirements

- Desktop videophone applications
  - channel capacity (e.g., p = 1) = 64Kbps
  - QCIF at 15 frames/s still requires 4.7Mbps
  - required compression ratio is 4.7Mbps/64Kbps = 73 !!
- Video conferencing applications
  - channel capacity (e.g., p=10) = 640Kbps
  - CIF at 30 frames/s requires 37Mbps
  - required compression ratio is 37Mps/640Kbps = 58 !!
- Q: How much compression does JPEG give?

Video coding algorithm

- combines intra-frame and inter-frame coding
- fast processing for on-the-fly video compression and decompression



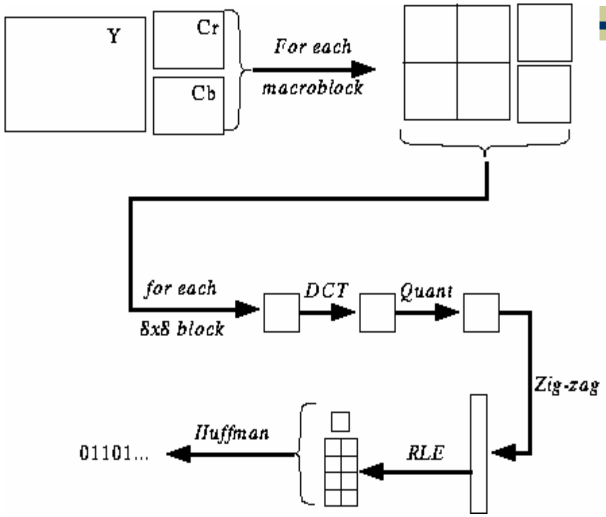
Video coding

- Video formats are CIF or QCIF images with 4:2:0 sub-sampling.
- 2 frame types: intra-coded frames (*I-frames*) and Predictive frames (*P-frames*).
- Algorithm begins by coding an I-frame using a JPEG-like method.

Video coding

- Each subsequent P-frame is encoded in terms of its predecessor using predictive inter-frame coding.
- At least one of every 132 frames is coded as an I-frame to provide an accessing point and as a reference image for accurate decoding.

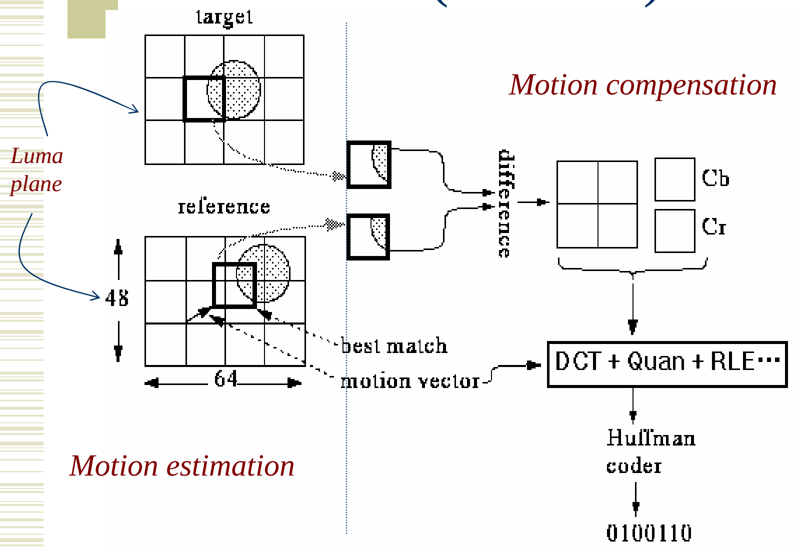
Intra-frame (I-Frame) coding



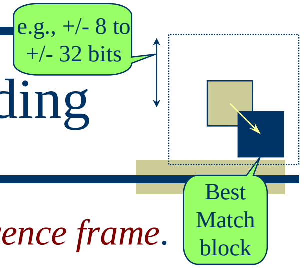
I-Frame coding

- Macro-blocks are 16×16 pixels on Y' plane of original image. It consists of 4 Y' (8×8) blocks, 1 C_B block and 1 C_R block.
- a constant quantizer value for all DCT coefficients.

Inter Frame (P-Frame) coding



P-Frame coding



- Previous frame is called the *reference frame*.
- The frame to be coded is called the *target frame*.
- Inter-frame coding is based on prediction for *each macro-block*. We compare the reference macro-block against the target macro-block.

P-Frame coding

- *Motion estimation*. Determine the motion vector, i.e., the relative position of the reference macro-block with respect to the target macro-block, using some matching function.
- *Motion compensation*. The difference between the 2 macro-blocks (if > certain threshold) is calculated and then sent to a JPEG-like encoder. If the difference < threshold, simply record the motion vector.

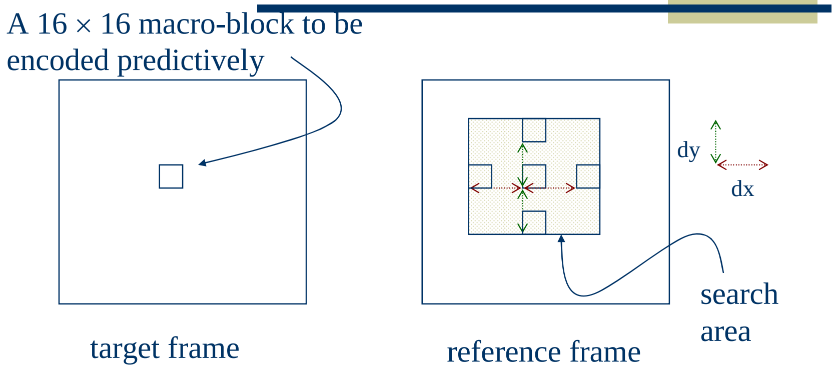
P-frame coding

- In most cases, predictive coding only makes sense for parts of the image and not for the whole image ⇒ not every macro-block in a P-frame is encoded using prediction. Some of them are encoded in the I-frame style.
- Since the motion vectors of adjacent macro-blocks often differ only slightly, only the differences of the motion vectors are encoded.

Block matching

- most time consuming part of the encoding process (target of optimization)
- takes place only on the luma component of frames
- The search is usually restricted to a small search area centered around the position of the target macro-block.
- The maximum displacement is specified as the maximum number of pixels in the horizontal and the vertical directions.

Search area



Search area

- If the maximum displacements in the horizontal and vertical directions are *dx* and *dy*, then
  - the search area = $(2dx + 16) \times (2dy + 16)$;
  - the number of candidate blocks = $(2dx + 1)(2dy + 1)$.
- Considering every candidate macro-block in the search area as a potential match is known as an *Exhaustive Search*, *Brute Force Search*, or *Full Search*.

Matching criteria

- A distortion function is used to quantify the similarity between the target macro-block and the candidate macro-blocks.
- The distortion function should be easy to compute and should result in good matches.
- Mean Absolute Difference (MAD)
  - most popular

$$\frac{1}{256}\sum_{p=1}^{16}\sum_{q=1}^{16}\left|A[p,q]-B[p,q]\right|$$

macro-block in target frame — macro-block in reference frame

Matching criteria

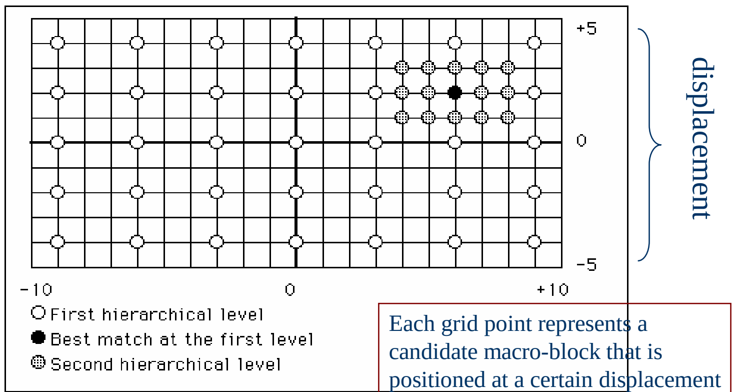
- Mean Square Difference (MSD) $\frac{1}{256}\sum_{p=1}^{16}\sum_{q=1}^{16}(A[p,q]-B[p,q])^2$
  - results in slightly better matches
- Pel Difference Classification (PDC)
  - compares the target macro-block and the candidate macro-block pixel by pixel
  - A pixel pair is a match if the difference < certain threshold *t*.
  - The greater the number of matching pixels, the better the match.

$$\sum_{p=1}^{16}\sum_{q=1}^{16}\left[\left(\left|A[p,q]-B[p,q]\right|\le t\right)?1:0\right]$$

Motion estimation algorithms

- Principle of Locality
  - Very good matches, if they exist, are likely to be found in the neighborhood of other good matches.
  - Example: Two-level hierarchical search:
first examine a number of sparsely spaced candidate macro-blocks from the search area and choose the best match as the center of a second, finer search.

Hierarchical search

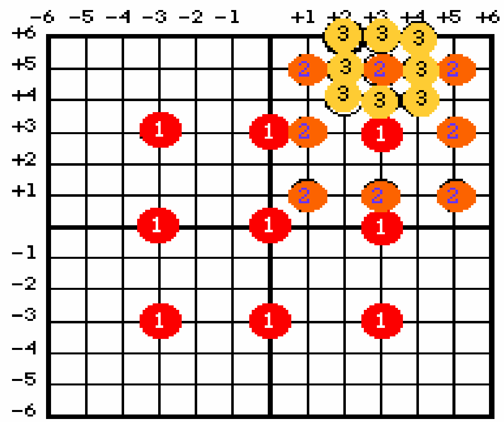


Each grid point represents a candidate macro-block that is positioned at a certain displacement from the center of the search area.

Three-step search

- Three Step Search (*TSS*)
  - Given a maximum displacement *d*, set step size $s = d/2$.
  - Given a center $[c_x, c_y]$, test nine points:
    - $[c_x$ +/- 0 or s, c_y +/- 0 or $s]$.
  - Take best match as new center, $s = \lceil s/2 \rceil$, repeat until s=1.
  - The first description of *TSS* uses a maximum displacement of +/- 6, hence the name.

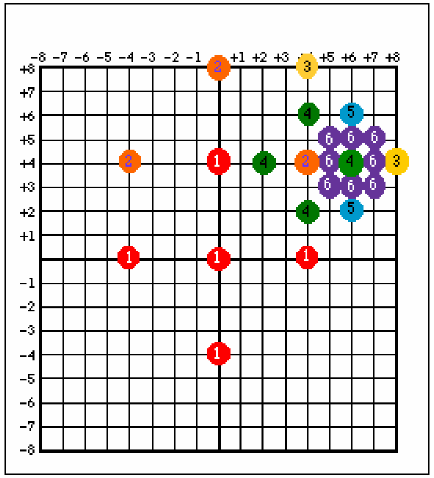
Three-step search



Two Dimensional Logarithmic Search (*TDL*)

- Given a center $[c_x, c_y]$, test 5 points:
  - $[c_x, c_y]$ & $[c_x$ +/- s, c_y +/- $s]$.
- If $[c_x, c_y]$ is the best match, $s = s/2$, repeat;
- else if $[a, b]$ is the best match, take it as the new center and repeat.
- If $s = 1$, then all nine points around the center are tested. Take best match.

Two Dimensional Logarithmic Search (*TDL*)



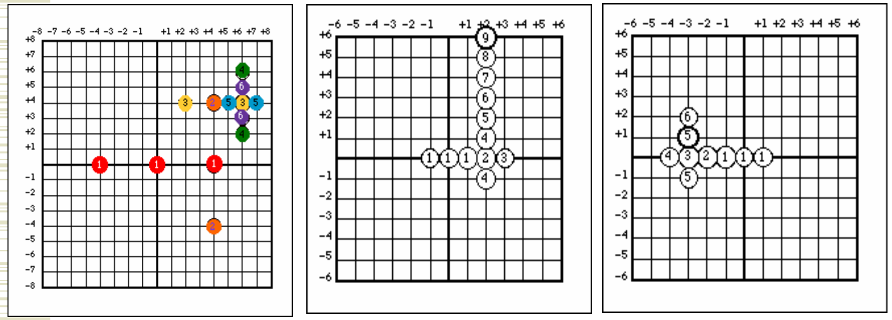
Orthogonal Search Algorithm (*OSA*)

- Given a center $[c_x, c_y]$, test 3 points:
  - $[c_x, c_y]$, $[c_x\text{-}s, c_y]$, $[c_x\text{+}s, c_y]$.
- Let $[a, b]$ be the best match, test 3 points:
  - $[a, b]$, $[a, b\text{+}s]$, $[a, b\text{-}s]$.
- Take best match as new center, set $s = s/2$, repeat.

One at a Time Search (*OTS*)

- Locate the best match on the horizontal axis. Then starting with this point, find the best match in the vertical direction.

OSA & *OTS*

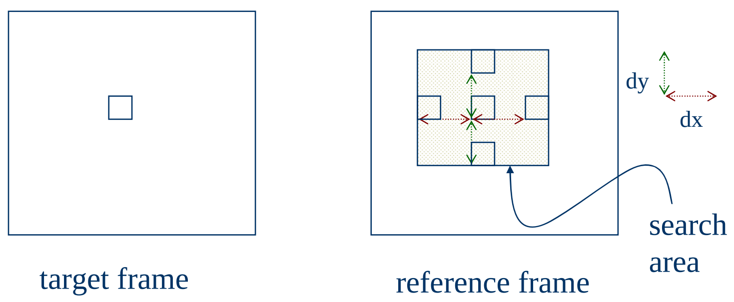


OSA *OTS*

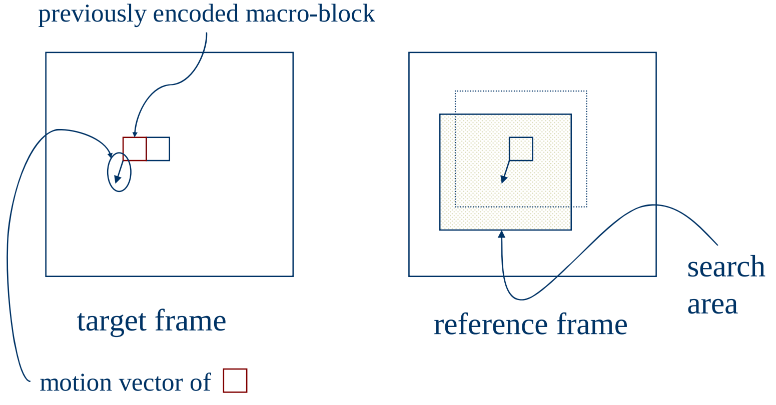
Dependent algorithms

- Observation: the closer the best matching macro-block is to the center of the search area, the faster are the algorithms.
- Based on the assumption that motion of adjacent (spatial and temporal) macro-blocks are correlated.
- Use the motion vectors of neighboring macro-blocks to calculate a prediction of the target macro-block’s motion, and this prediction is used as a center of the search.

Search area (without prediction)

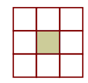


Search area (with prediction)



Dependent algorithms

- Spatial dependency
  - Take a weighted average of the neighboring macro-blocks' motion vectors.
  - Q: can we use all 8 neighbors? A: Nah!
  - The order in which macro-blocks are matched restricts the choice of neighboring macro-blocks that can be used.

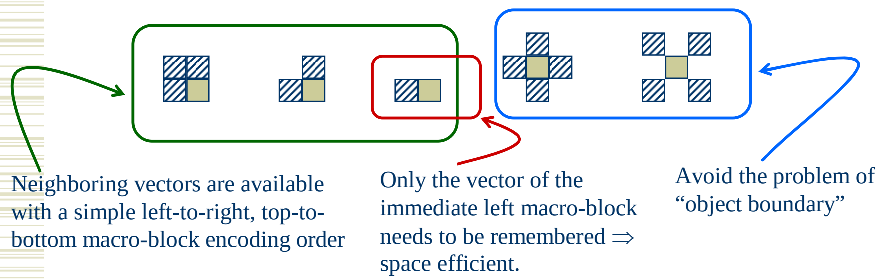


Dependent algorithms

- Can you think of the advantages and disadvantages of using the neighbors (▨) as shown below?

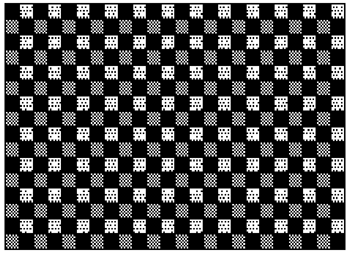
Dependent algorithms

- Can you think of the advantages and disadvantages of using the neighbors (▨) as shown below?



Dependent algorithms

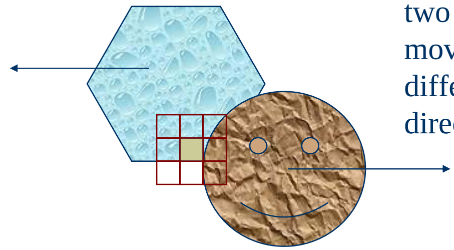
- Example. A multi-pass prediction process:



Dark boxes indicate target macro-block and lighter boxes represent the neighbors whose motion vectors assist the matching algorithm.

Dependent algorithms

- If a macro-block falls on an object boundary, the motion vectors of its neighboring blocks may carry conflicting values.

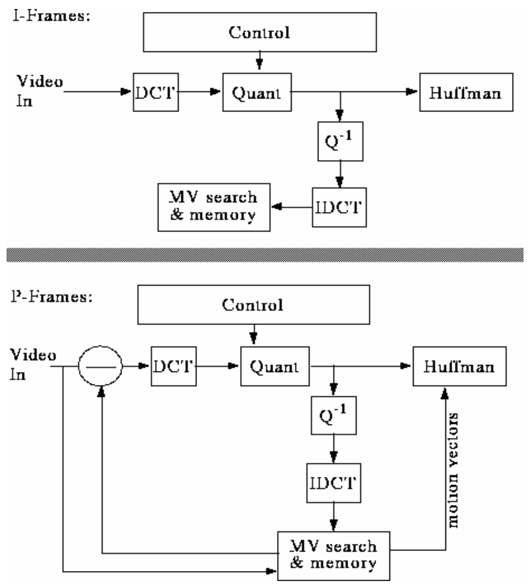


two objects moving in different directions

Dependent algorithms

- To circumvent the problem of object boundaries, we can take a voting approach instead of averaging the neighbors' motion vectors.
- If the motion vectors of the neighboring blocks are not sufficiently uniform then the search for the target block might be carried out as normal, as though no spatial dependency was being exploited.

Logic Diagram of H.261 Codec



Constant bit rate

- The bit rate required depends on the type of frame used and also the complexity of the video:
  - I-frame: higher bit rate; P-frame: lower bit rate
  - motion-intensive: higher bit rate; static: lower bit rate
- Need a buffer to regulate the traffic

A feedback mechanism

- The level of quantization depends on the amount of buffer space left. Less space ⇒ coarser quantization ⇒ lower data rate ⇒ lower picture quality. This mechanism enforces a constant data rate at the output of the coder.

MPEG

- What is MPEG?
  - A standard for delivery of audio and motion video.
  - By the Motion Picture Expert Group, ISO activity in 1993.
  - Official name: WG11 of JTC 1 / SC 29.
  - Further developments lead to standards of MPEG-2, MPEG-4, MPEG-7.
  - MPEG-1 targeted at VHS-quality video on CD-ROM, i.e., about 352×240×30fps video + CD audio at (up to) 1.5Mbps.

    352×288×25fps for PAL

MPEG

- MPEG-2 for higher resolution, CCIR 601 digital television quality video (720 × 480 × 30fps) @ 2-10 Mbps. MPEG-2 supports interlaced video format, scaleable video coding for a variety of applications which need different image resolutions.
- MPEG-3 for HDTV-quality video @ 40Mbps. Since MPEG-2 can be scaled to cover HDTV applications, MPEG-3 was dropped.

MPEG

- Standard has 3 parts:
  - Video: based on H.261 and JPEG, optimized for motion-intensive video applications.
  - Audio: based on MUSICAM technology
    - 64/128/192 kbps per channel
    - compression ratio: 5:1 to 10:1
  - System: control interleaving of streams, synchronization.

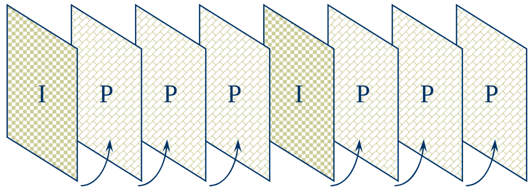
MPEG encoding features

- 4:2:0 sub-sampling (main profile)
- random access via I-frames
- fast forward/reverse searches
- reverse playback
- suitable for asymmetric compression.
  - Electronic publishing, games and entertainment require compression once and frequent decompression.

not needed for H.261 (applies to: random access via I-frames; fast forward/reverse searches; reverse playback)

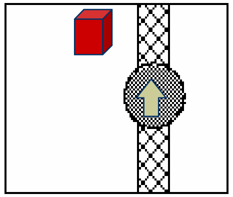
Temporal dependency in MPEG sequences

- Recall H.261 dependencies:

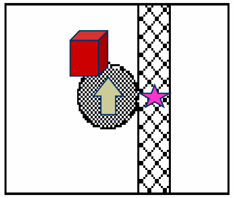


Temporal dependency

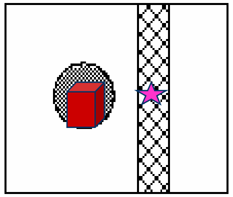
- Prediction for the P-frames sometimes takes the advantage of bi-directional prediction.
  - For instance, the target image in the following takes both the previous and the future references for its derivation.



previous



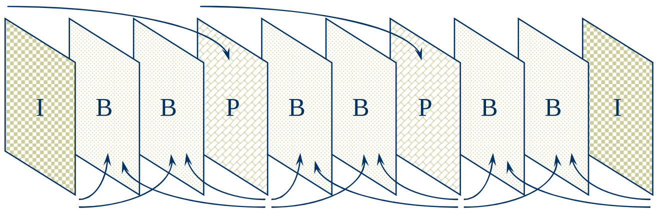
target image



future

Temporal dependency

- MPEG uses another frame type: B-frame, which is similar to P-frame, but prediction is based on a previous as well as a future frame.



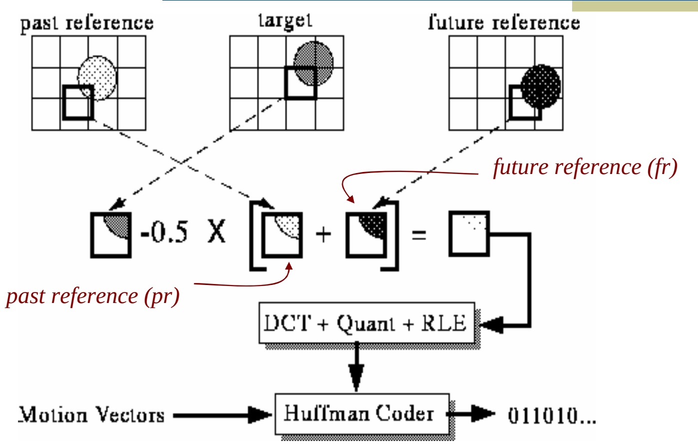
MPEG frame types

- I-frames (Intra-coded frames)
  - use JPEG for I-frame encoding.
  - lowest compression as no temporal redundancy exploited.
  - provide points for random access in an MPEG stream.
- P-frames (Predictive-coded frames)
  - require previous I or P frame for encoding and decoding

MPEG frame types

- B-frames (Bi-directionally predictive-coded frames)
  - encode the motion vector and difference of prediction based on the previous and the following I or P frames
  - can use forward, backward prediction, or interpolation
  - generally achieve a higher compression than I or P frames
  - Two motion vectors are used (forward and backward). Interpolation of two reference macro-blocks is “diffed” with the target macro-block.
  - never used as a reference frame (for the encoding of other frames).
  - Any disadvantages?

B-frame coding (interpolation)



B-frame encoding

- We compare target macro-block against the following 3 cases:
  - *pr*,
  - *fr*,
  - (*pr*+*fr*)/2
- Take the best match. If none gives a reasonably good match, revert to I-frame-like encoding for the target marco-block.

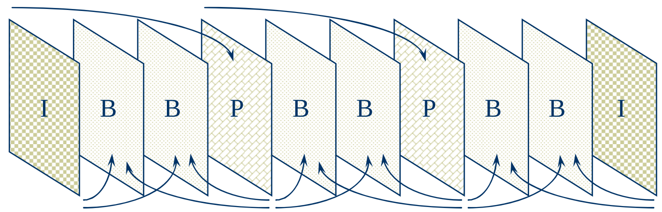
Choosing a frame type

- I-Frame:
  - good for direct access,
  - bad for compression.
- B-Frame:
  - best for compression,
  - adds delay to the encoding process,
  - more computationally expensive,
  - needs a lot of buffer space.
- P-Frame:
  - good for compression.

Choosing a frame type

- Typical frame type sequence is

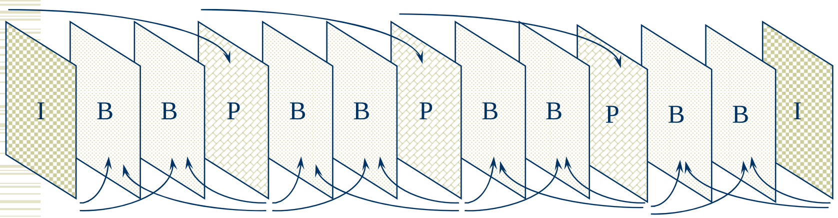
  ... IBBPBBPBBIBBPBBPBBI ...



Choosing a frame type

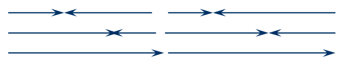
- Or …

  ... IBBPBBPBBIBBPBBPBBPBBI ...



Inter-frame coding

- Actual pattern is up to encoder, and need not be regular.
- Bi-directional prediction:
  - I B B P B B P



- Transmitting order: 1, 4, 2, 3, 7, 5, 6, ...

Relative performance

- Compression performance (example) of different frame types

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