Introduction to H.264

Bo Hong

Multimedia Communications Laboratory

University of Texas at Dallas

November 22, 2002

Outline

- History of video coding standards
- Overview of H.264
- Selected new features in H.264
  - Video coding
    - Integer transform
    - Intra prediction
    - De-blocking filter
  - Error resilience
    - Multiple frame prediction
    - SP/SI frame
- Performance of H.264

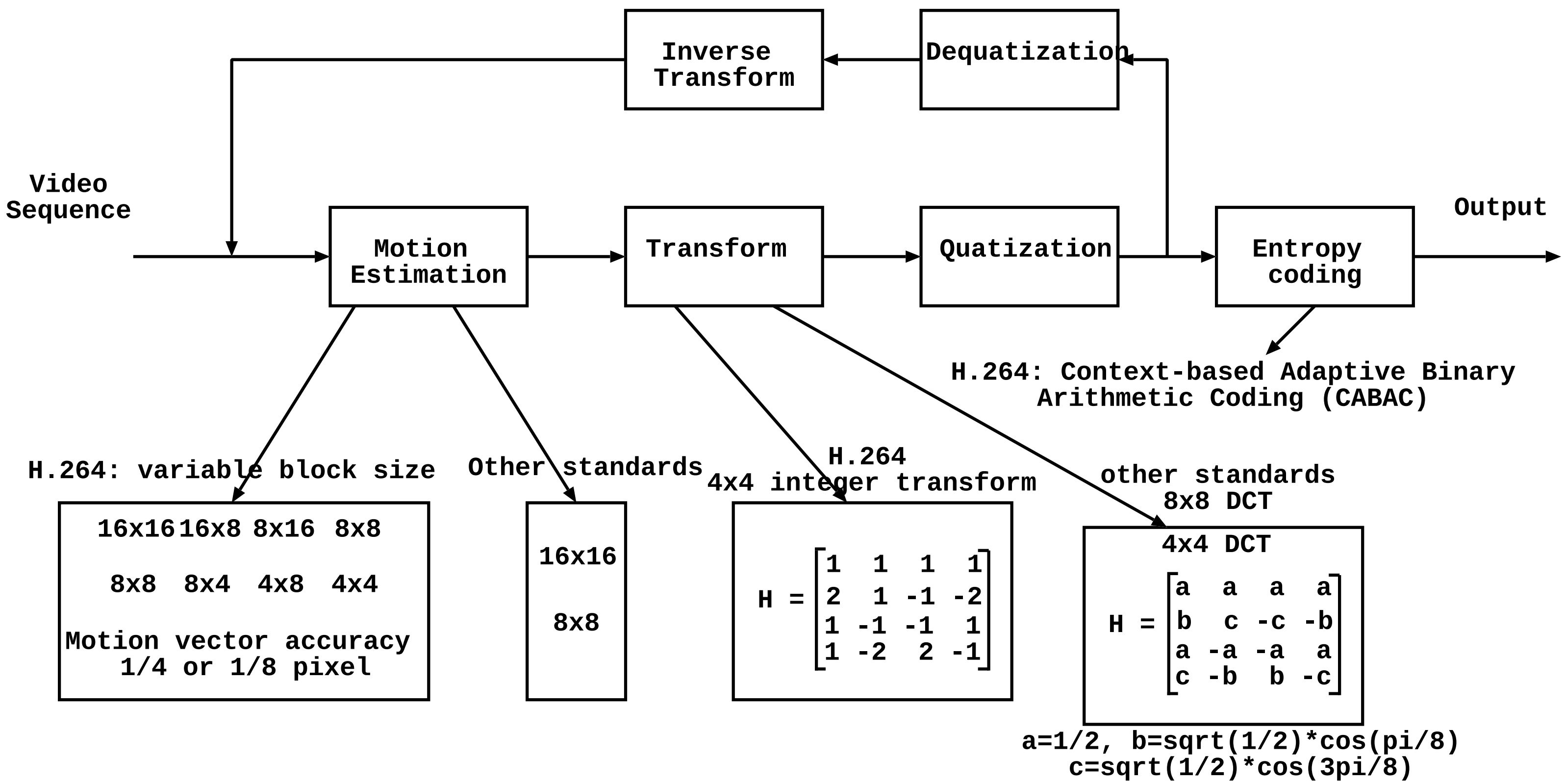
History of Video Coding Standards

- ITU-T Video Coding Experts Group (VCEG)
  - H.261 (1990) → H.263 (1995) → H.263+ (1998) → H.26L
- ISO Motion Picture Experts Group (MPEG)
  - MPEG1(1991) → MPEG2 (1994) → MPEG4 (1999)
- Joint Video Team (JVT) (VCEG/MPEG) 2001
  - H.264/MPEG4 part 10. Official title: Advanced Video Coding (AVC)
  - International standard: December 2002

Overview of H.264

- Video coding is similar in spirit to other standards but with important differences
- Key new features:
  - Enhanced motion estimation with variable block size
  - Integer block transform
  - Improved in-loop deblocking filter
  - Enhanced entropy coding
- Average bit rate reduction of 50% given fixed fidelity compared to any other standard
- Complexity vs. Coding efficiency

Video Coding System



Integer Transform

- Motivation
  - Quantization reduces precision
  - Small block size (4x4) reduces the performance loss
- Discrete Cosine Transform (DCT) has been used widely in standards
  - Good energy compact property
  - Orthogonal bases

From DCT to Integer Transform

$$Y = AXA^T = \begin{bmatrix} a & a & a & a \\ b & c & -c & -b \\ a & -a & -a & a \\ c & -b & b & -c \end{bmatrix} X \begin{bmatrix} a & b & a & c \\ a & c & -a & -b \\ a & -c & -a & b \\ a & -b & a & -c \end{bmatrix}$$

$$Y = \left(CXC^T\right) \oplus E = \left(CXC^T\right) \oplus \begin{bmatrix} a^2 & ab & a^2 & ab \\ ab & b^2 & ab & b^2 \\ a^2 & ab & a^2 & ab \\ ab & b^2 & ab & b^2 \end{bmatrix}$$

- where $C = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & d & -d & -1 \\ 1 & -1 & -1 & 1 \\ d & -1 & 1 & -d \end{bmatrix}$
- $a = 1/2,\ b = \sqrt{1/2}\cos\pi/8,\ c = \sqrt{1/2}\cos 3\pi/8,\ d = c/b \approx 0.414$

Make the Integer Transform Orthogonal

- To make the transform simple, we approximate d with $1/2$
- To maintain orthogonality, modify b to $b = \sqrt{2/5}$
- Now the transform becomes

$$Y = (HXH^T) \oplus \begin{bmatrix} a^2 & ab & a^2 & ab \\ ab & b^2 & ab & b^2 \\ a^2 & ab & a^2 & ab \\ ab & b^2 & ab & b^2 \end{bmatrix}$$

- where $H = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1/2 & -1/2 & -1 \\ 1 & -1 & -1 & 1 \\ 1/2 & -1 & 1 & -1/2 \end{bmatrix}$, $b = \sqrt{2/5}$

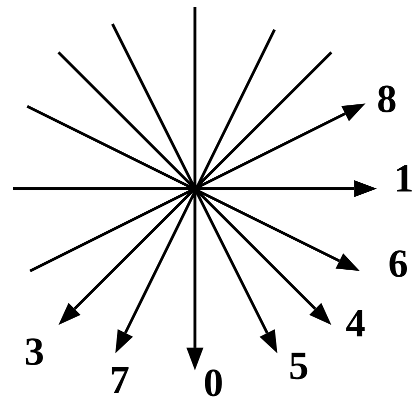
Performance of Integer Transform

- Scenario: Stationary Gauss-Markov input with correlation coefficient $\rho = 0.9$
  - Coding gain for H: $5.38dB$, Coding gain for DCT: $5.39dB$
- Video compression: input are prediction residuals, $\rho < 0.9$

Intra Prediction

- Directional spatial prediction

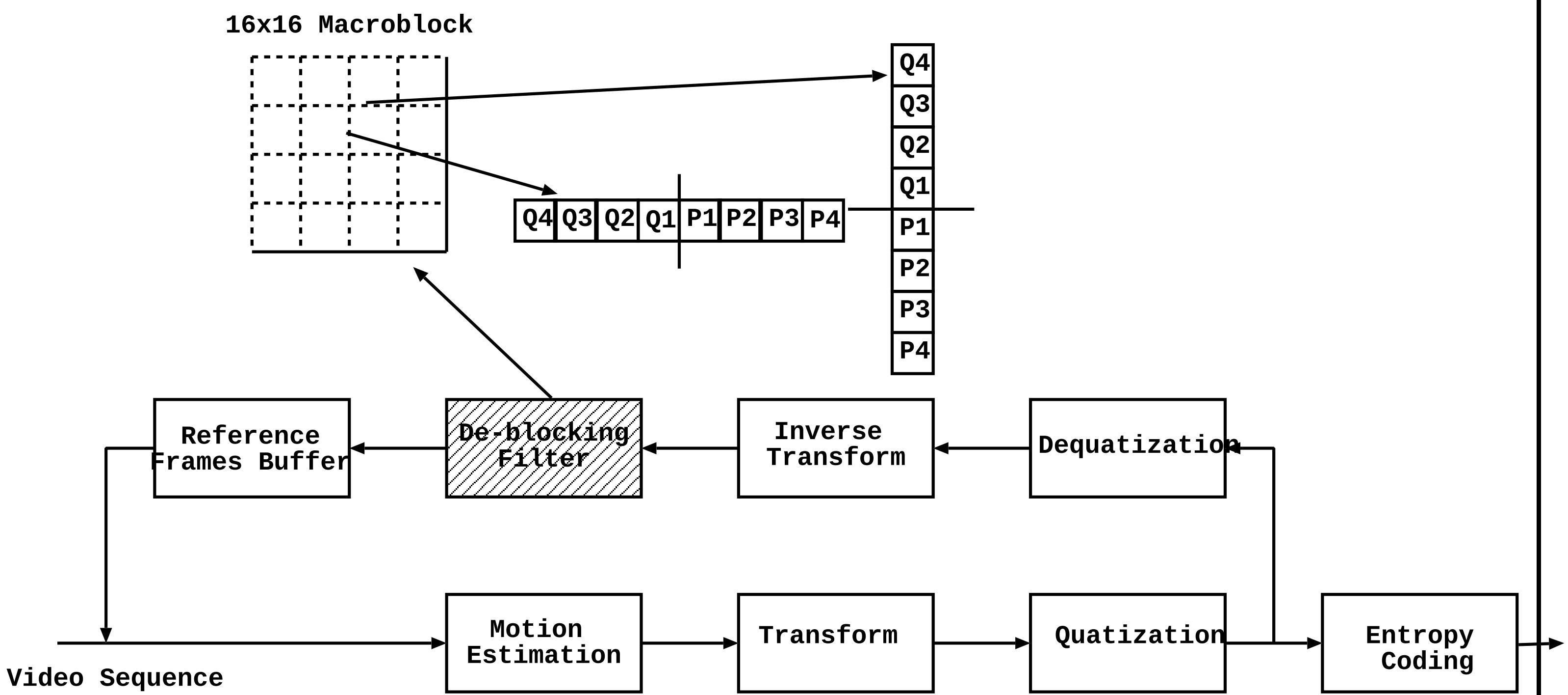
M	A	B	C	D	E	F	G	H
I	a	b	c	d				
J	e	f	g	h				
K	i	j	k	l				
L	m	n	o	p				



- 9 optional prediction modes
- $a - p$ is predicted from $A - M$
- For example: $a = (A + 2M + I + 2)/4$ (Mode 3)

De-blocking Filter

- Motivation: Reduce blocking artifacts



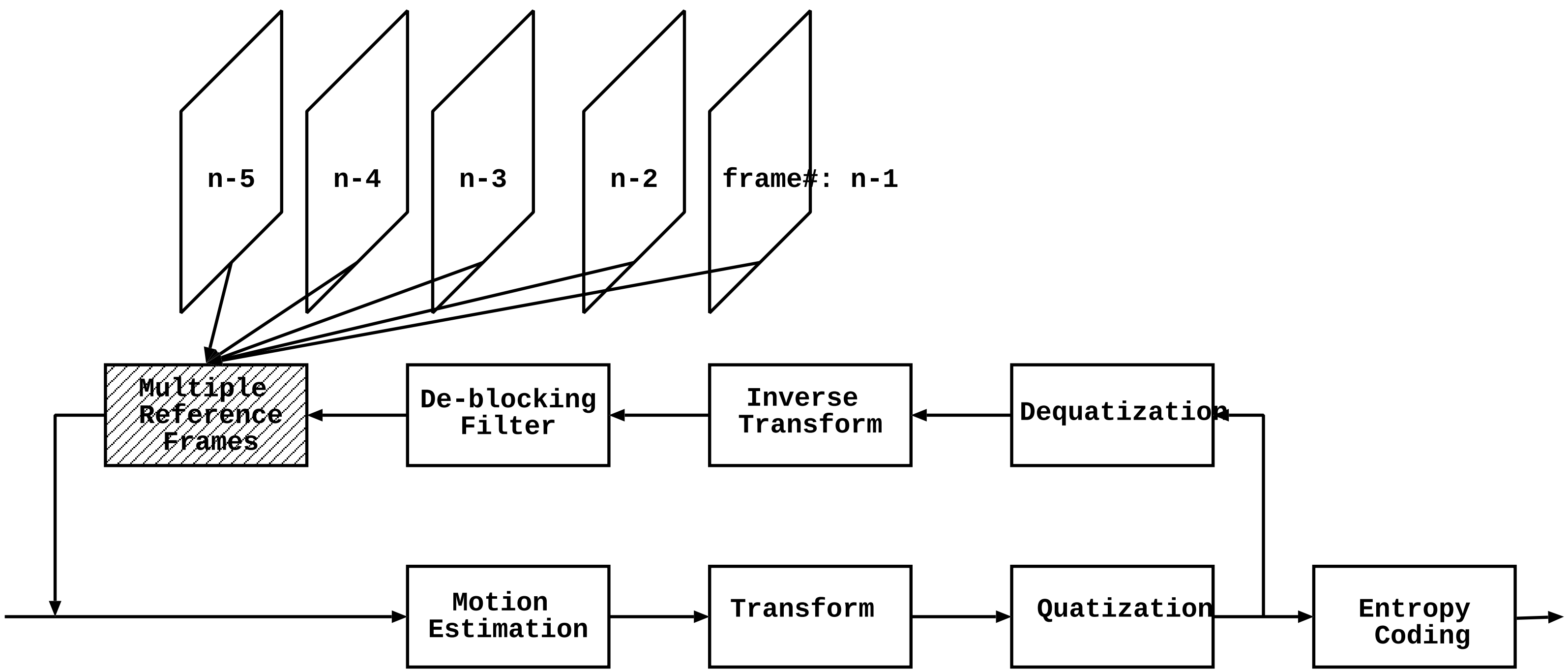
De-blocking Filter (Cont.)

- Improves subjective visual quality of the decoded picture.
- Much better than post filtering.
- Highly content adaptive filtering
  - On edge level, filtering strength is dependent on the code mode, motion vector, and values of residuals
  - On sample level, quantizer dependent thresholds can turn off filtering for any individual sample

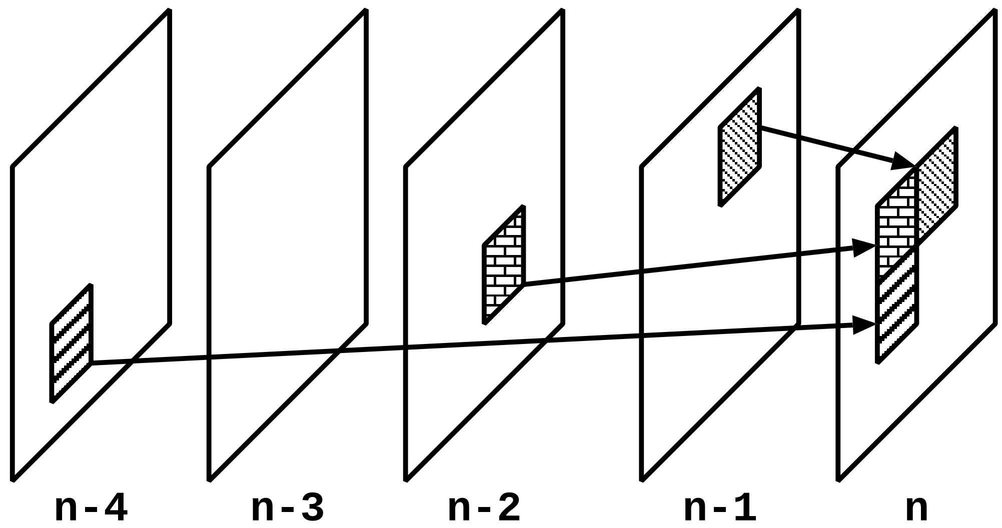
Error Resilience Features in H.264

- **Multiple reference frames**
- **SP/SI frames**
- Slice structured coding
- Data partitioning
- Intra frame or intra microblock updates

Multiple Reference Frames



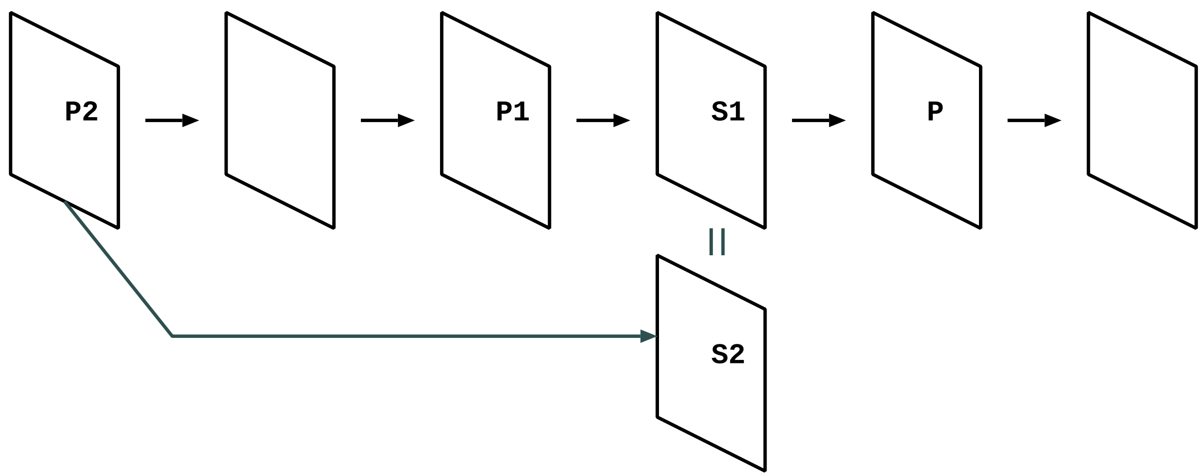
A Scenario using Multiple Reference Frames



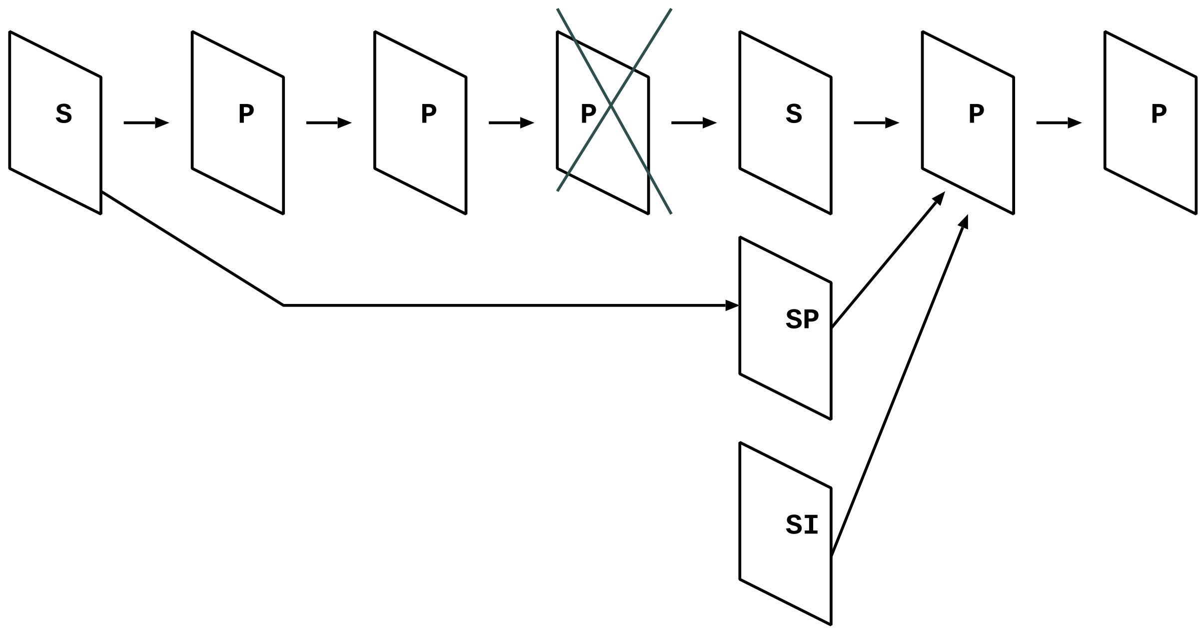
- Improve the performance of motion estimation
- Solve the uncovered background problem
- Increase the coding and decoding complexity
- Increase the size of reference frames buffer

S Frame

- Idea
  - Reconstruct identical frame even when different reference frames were used

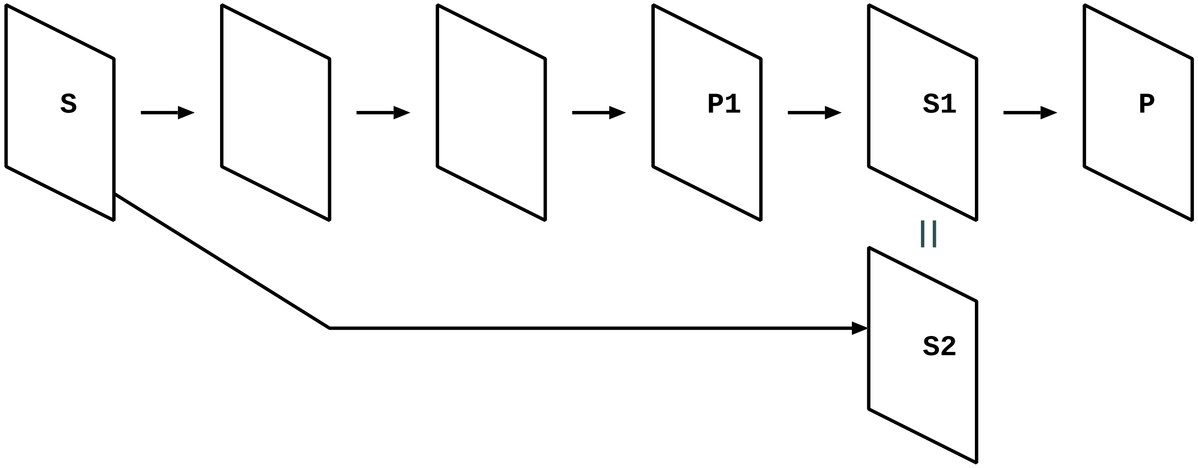


S Frame in Error Resilience



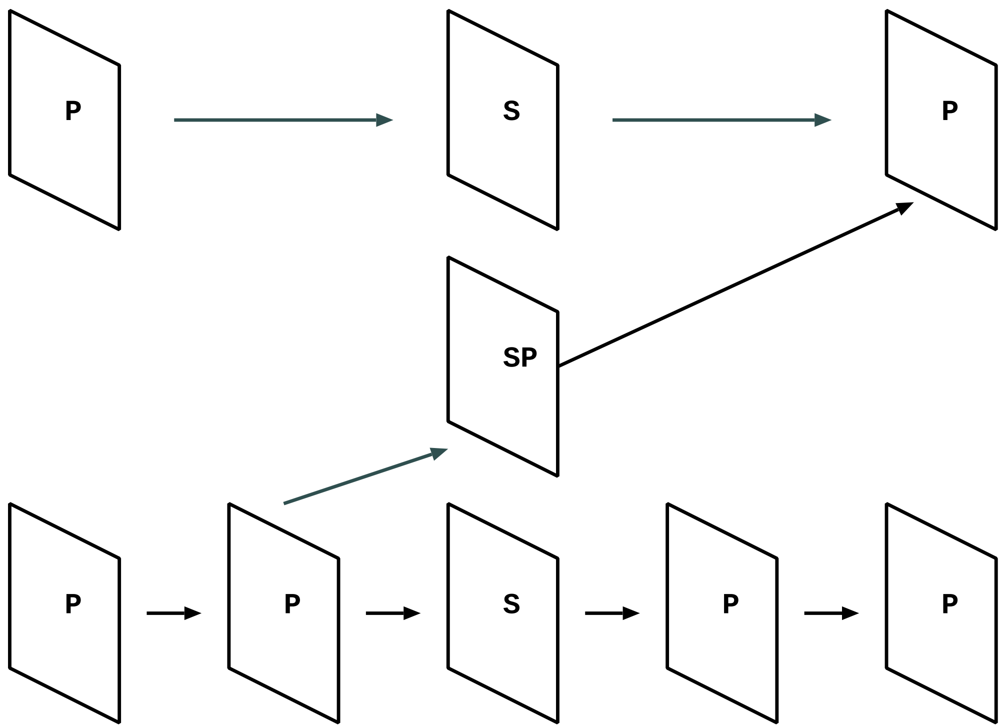
- Only one of S/SP/SI frames will be sent
- SP frame will be sent when its reference frame is available
- Problem: S/SP/SI will be used as reference frame for the following frames, how to make them identical?

How S Frame Works



- $S_1 = L_{pred1} + L_{err1}$, $S_2 = L_{pred2} + L_{err2}$
- To make $S_1 = S_2$, we have
  - $L_{err2} = S_1 - L_{pred2}$

S Frame in Fast Forward



- SP frame will be sent only during bitstream switching
- More efficient than using I frame

Performance of H.264

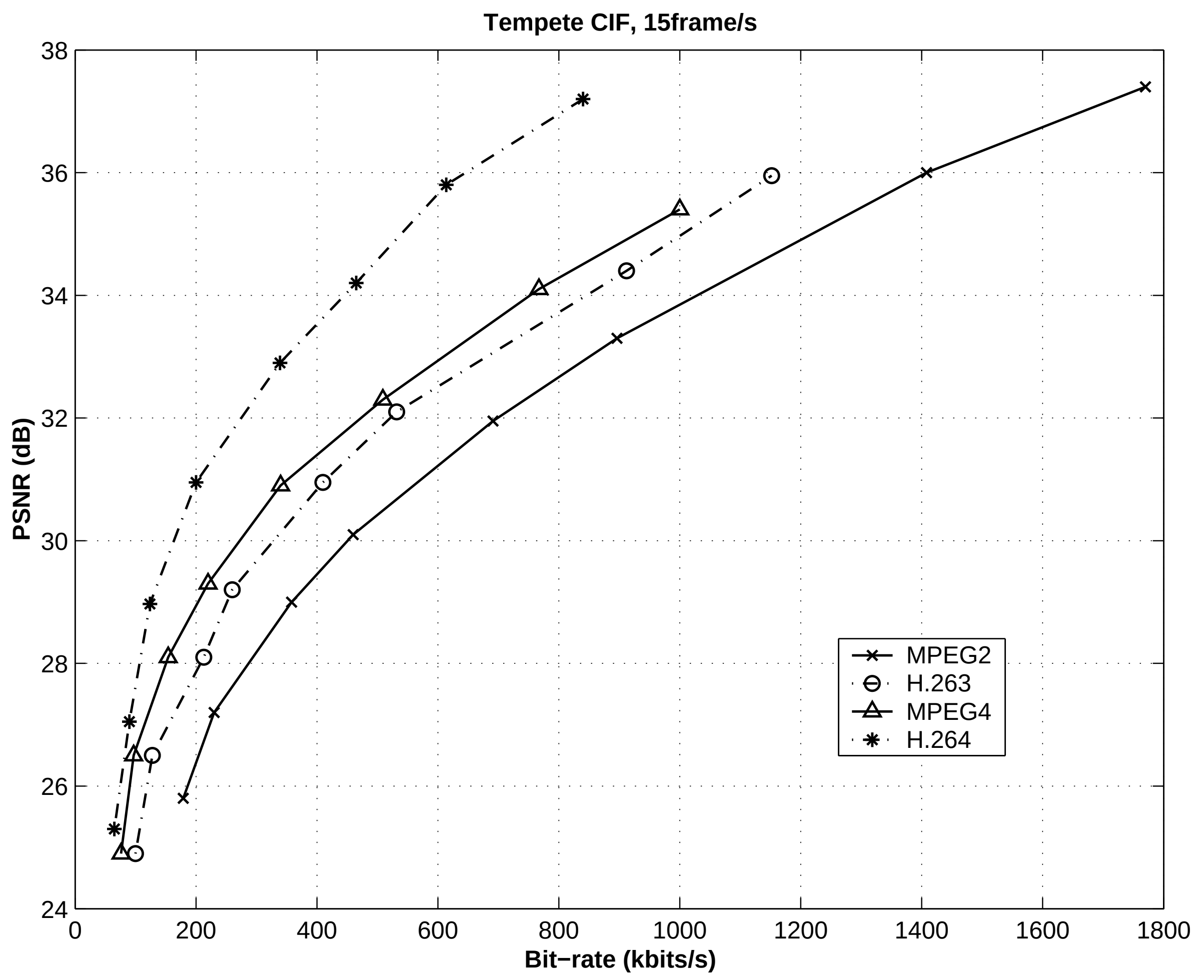
- Results from [3]
- Using same rate-distortion optimization techniques for all codecs
- Several video sequences for each test
- Compare four codecs:
  - MPEG2
  - H.263 (baseline profile)
  - MPEG4 (simple profile)
  - H.264

Rate Saving Compare to Other Standards

Table 1: Average rate savings compare to other standards

Codec	**MPEG4**	**H.263**	**MPEG2**
H.264	39%	49%	64%
MPEG4	-	17%	43%
H.263	-	-	31%

Performance Comparison



Reference

1. Special session of H.26L, ICIP 2002
2. Ftp site: http://bs.hhi.de/~suehring/tml/
3. The Emerging JVT/H.26L Video Coding Standards, Proceeding of IBC 2002, Amsterdam, NL, Sept. 2002