Advanced Hybrid Video Coding
Multiple Reference Pictures

Motion-compensated prediction with multiple reference pictures
- Store multiple reconstructed in a decoded picture buffer
  - Sliding window buffer (store most recently decoded pictures)
  - Adaptive buffer management (transmit commands)
- Construct reference picture list for current picture
  ➤ Choose reference picture in addition to motion vector
  ➤ Transmit reference picture index in addition to motion vector
Reference Picture Selection

Typical approach

- Determine motion vector $m_r$ for each considered reference picture $r \in R$
- Select motion parameters $\{r, m_r\}$ among the pre-determined sets

Criterion for reference picture selection

- Lagrangian decision similar to motion search
- Choose motion parameters $\{r, m_r\}$ according to

$$r^* = \arg \min_{r \in R} D_k(r, m_r) + \lambda_M R_k(r, m_r)$$

with

$D_k(r, m_r)$ – Distortion between original block $s$ and prediction signal $\hat{s}$

$R_k(r, m_r)$ – Number of bits for reference index $r$ and motion vector $m_r$

→ Usually: Same distortion measure as for sub-sample search
Multiple Reference Pictures — Coding Efficiency

Coding experiment with H.265 | MPEG-H HEVC

- IPPP coding: Use $N$ previous pictures as reference pictures
- Vary number $N$ of available reference pictures from 1 to 8
- Reference: Single reference picture
→ Coding efficiency increases with number of reference pictures
Multi-Hypothesis Prediction

Motion-compensated prediction with multiple motion hypotheses

- Weighted summation of multiple displaced reference blocks $s_{r_k}'(x + m_k)$
- Simplest and most commonly used variant

$$\hat{s}[x] = \frac{1}{K} \sum_{k=0}^{K-1} s_{r_k}'(x + m_k)$$
Multi-Hypothesis Prediction

Motivation of multi-hypothesis prediction

- **Idealized assumption:**
  Individual prediction errors \( u_k[x] = s[x] - s'_r(x + m_k) \) are uncorrelated

- Reduced residual power spectrum compared to single-hypothesis prediction

  \[
  \Phi_{UU}(\omega) \approx \frac{1}{K} \Phi_{u_0u_0}(\omega)
  \]

- Improved coding efficiency for residual signal

- But: Bit rate required for transmitting motion data is increased

Block adaptive selection of number of motion hypotheses

- Multi-hypothesis prediction does not improve coding efficiency for all blocks

- Adaptive selection of number of motion hypotheses

- Video coding standards: Only up to two motion hypotheses (bi-prediction)

- Motion estimation: Iterative refinement of motion vectors
Multi-Hypothesis Prediction in Video Coding Standards

H.262 | MPEG-2 Video, H.263, MPEG-4 Visual

- I (intra only), P (uni-prediction), and B (bi-prediction) pictures
- B pictures are only supported as part of an BBP or BBI group
- B pictures support the following basic MCP modes
  - **Forward**: Uni-prediction from preceding I/P picture
  - **Backward**: Uni-prediction from succeeding I/P picture
  - **Bi-directional**: Bi-prediction from preceding and succeeding I/P picture
Multi-Hypothesis Prediction in Video Coding Standards

H.264 | MPEG-4 AVC and H.265 | MPEG-H HEVC
- Generalized concept of bi-prediction and multiple reference pictures
- Decoupling of coding order and picture types
- I, P, and B slices instead of I, P, and B pictures
- B slices: Two reference picture lists (list 0 and list 1)
  - Both lists can include pictures from past and future
  - Each stored picture can be included in either or both lists
- Allows new (more efficient) coding structures (discussed later)

Basic inter-picture coding modes supported in B slices
- **List 0 prediction:** Uni-prediction with picture from list 0
- **List 1 prediction:** Uni-prediction with picture from list 1
- **Bi-prediction:** Bi-prediction with picture from list 0 and picture from list 1
  - Both hypotheses can use the same picture
  - Enables bi-prediction without modifying the coding order (IBBB)
Motion Estimation for Multi-Hypotheses Prediction

Motion estimation for multiple motion vectors

- Need to estimate multiple motion vectors for a block in current frame
- Independent estimation is sub-optimal
- Estimation in product space is too complex

Independent estimation of motion hypotheses is not optimal

- Example: SSD distortion for bi-prediction

\[
D_{\text{Bi}} = \sum_{x, y} \left( s[x, y] - \frac{1}{2} (\hat{s}_1[x, y] + \hat{s}_2[x, y]) \right)^2 \\
= \frac{1}{4} \sum_{x, y} \left( (s[x, y] - \hat{s}_1[x, y]) + (s[x, y] - \hat{s}_2[x, y]) \right)^2 \\
= \frac{1}{4} D_1 + \frac{1}{4} D_2 + \frac{1}{2} \sum_{x, y} (s[x, y] - \hat{s}_1[x, y]) (s[x, y] - \hat{s}_2[x, y]) \\
\]

→ Minimization of \( D_1 \) and \( D_2 \) does not minimize \( D_{\text{Bi}} \)
Iterative Motion Estimation for Multi-Hypotheses Prediction

Iterative estimation for multiple motion hypotheses (example: bi-prediction)

- Distortion for bi-prediction can be written as

\[
D_{Bi} = \sum_{x,y} \left| s[x, y] - \frac{1}{2} \left( \hat{s}_1[x, y] + s_{ref}^{(2)}(x - m_x^{(2)}, y - m_y^{(2)}) \right) \right|^{\beta}
\]

\[
= \frac{1}{2^\beta} \cdot \sum_{x,y} \left| (2 \cdot s[x, y] - \hat{s}_1[x, y]) - s_{ref}^{(2)}(x - m_x^{(2)}, y - m_y^{(2)}) \right|^{\beta}
\]

\[
= \frac{1}{2^\beta} \cdot \sum_{x,y} \left| s^*[x, y] - s_{ref}^{(2)}(x - m_x^{(2)}, y - m_y^{(2)}) \right|^{\beta}
\]

→ Conventional motion search, but with modified original signal \( s^*[x, y] \)

Iterative algorithm for bi-prediction

1. Independent estimation of first motion hypothesis
2. Conditional estimation of second/first motion hypothesis (alternately)
3. Repeat last step until convergence

→ Algorithm can be extended to more than two hypotheses
Multi-Hypothesis Prediction — Coding Efficiency

Coding experiment with H.265 | MPEG-H HEVC, IPPP/IBBB coding structure

- Block-adaptive bi-prediction with different number of reference pictures
- Reference: Uni-prediction with same number of reference pictures
  - Significant coding gain for block-adaptive bi-prediction
  - Bi-prediction gain slightly increases with number of reference pictures
Multi-Hypothesis Prediction — Coding Efficiency

Coding experiment with H.265 | MPEG-H HEVC, IPPP/IBBB coding structure
- Average bit rate savings relative to uni-prediction with single reference picture
- Average bit rate savings for two classes of HD test sequences
  ➡ Bit-rate savings increase with number of reference pictures
  ➡ Significant coding gains by combining multi-hypothesis prediction and multiple reference pictures
Conventional Coding Structures

IPPP / IBBB coding
- All pictures are coded in acquisition/display order
- Pictures can be coded as P/B pictures (or P/B slices)
- Usage of multiple reference pictures can be enabled

Conventional B pictures
- Pictures are coded using BBP or BBI groups
- P pictures can be replaced with B pictures
- Usage of multiple reference pictures can be enabled
- Two hierarchy levels: I/P pictures & B pictures
Hierarchical B pictures

- Video sequence is partitioned into so-called groups of pictures (GOPs)
- Multiple hierarchy levels (most common design: dyadic structures)
- Key pictures: Pictures of lowest hierarchy level
- Reconstruction quality of hierarchy levels have different importance
- Cascading of quantization parameters

\[ QP_k = QP_0 + \delta_1 + (k - 1) \]  
(typically: \( \delta_1 = 4 \))
Hierarchical Coding Structures — Coding Efficiency

IBBB

\[ \begin{array}{ccccccccccc}
I/B & B & B & B & B & B & B & B & B & B \\
\end{array} \]

GOP2

\[ \begin{array}{ccccccccccc}
I/B & B_1 & B_0 & B_1 & B_0 & B_1 & B_0 \end{array} \]

GOP4 (simple)

\[ \begin{array}{ccccccccccc}
I/B & B_1 & B_1 & B_1 & B_1 & B_0 & B_1 & B_1 & B_0 \end{array} \]

GOP4 (hier.)

\[ \begin{array}{ccccccccccc}
I/B & B_2 & B_1 & B_2 & B_1 & B_1 & B_2 & B_1 & B_0 \end{array} \]

GOP8 (hier.)

\[ \begin{array}{ccccccccccc}
I/B & B_3 & B_2 & B_3 & B_1 & B_3 & B_2 & B_3 & B_0 \end{array} \]

cascaded QP assignment with \( \delta_1=4 \)

average of entertainment-quality video content

\[ \begin{array}{ccccccccccc}
\text{GOP2} & \text{GOP4 (simple)} & \text{GOP4 (hier.)} & \text{GOP8 (hier.)} \\
\end{array} \]

\[ \begin{array}{ccccccccccc}
\text{ParkScene} & \text{BQTerrace} & \text{Average} \\
\text{Kimono} & \text{BasketballDrive} & \\
\end{array} \]

\[ \begin{array}{ccccccccccc}
\text{GOP2} & \text{GOP4 (simple)} & \text{GOP4 (hier.)} & \text{GOP8 (hier.)} \\
\end{array} \]

\[ \begin{array}{ccccccccccc}
\delta_1=4 & \delta_1=3 & \delta_1=2 & \delta_1=1 \\
\text{same QP} & \\
\end{array} \]

average of entertainment-quality video content

\[ \begin{array}{ccccccccccc}
\text{GOP2} & \text{GOP4 (simple)} & \text{GOP4 (hier.)} & \text{GOP8 (hier.)} \\
\end{array} \]

average of entertainment-quality video content

\[ \begin{array}{ccccccccccc}
\delta_1=4 & \delta_1=3 & \delta_1=2 & \delta_1=1 \\
\text{same QP} & \\
\end{array} \]
Hierarchical Coding Structures — Effect of QP Cascading

- **Same QP for all hierarchy levels**
  - Roughly same PSNR over all pictures (similar as for IBBB coding)

- **QP Cascading over hierarchy levels**
  - Higher PSNR for key pictures (nonetheless, video appears smooth)
  - PSNR losses for non-key pictures are outweighed by bit rate savings
Low-Delay Hierarchical Coding Structures

Low-delay hierarchical coding structures

- Low-delay applications: Pictures are coded in acquisition/display order
- Can still change reference picture lists & QP cascading
- Usage of pictures with higher quality (lower QP) often improves prediction quality for other pictures (outweighs additional bit rate)
- Coding efficiency can often be improved
In-Loop Filters

Filtering of reconstructed pictures

- Reduce visual impact of typical coding artifacts
- Blocking artifacts (block-wise prediction and transform coding)
- Ringing artifacts (long transform basis functions & interpolation filters)

Post filters or in-loop filters

- Both improve subjective quality of output pictures
- In-loop filters also improve prediction signal for following pictures
- In-loop filters are normative

In-loop filters

- Deblocking filter (H.263, H.264 | MPEG-4 AVC, H.265 | MPEG-H HEVC)
- Deringing filter (H.265 | MPEG-H HEVC ⇒ Sample adaptive offset)
- Adaptive Wiener filter (linear filter minimizing distortion)
Deblocking Filter — Basic Principle

Adaptive smoothing of 1D line segments across block boundaries

- Correction values $\Delta_0$, $\Delta_{p1}$, and $\Delta_{q1}$ depend on differences between samples values and coding parameters (QP, motion, intra/inter)
- Different filtering modes selected based on sample differences
- Stronger low-pass filtering for higher QP values (i.e., stronger quantization)
Deblocking Filter — Example

without deblocking filter

with deblocking filter
Deblocking Filter — In-Loop Filter vs Post Filter

Coding experiment with H.265 | MPEG-H HEVC

- Non-normative post filter vs normative in-loop filter
- Coding with hierarchical B pictures
  - In-loop filter also improves prediction signal for following pictures
  - In-loop filter provides larger coding gains
First operation mode: Edge offset mode

- **Goal:** Reduce ringing artifacts around edges
- **Select one of four edge classes for a CTU** (specifies edge direction)
- **Classify each sample into one of five categories**
- **Transmit offset for each category**
Sample Adaptive Offset (SAO) in H.265 | MPEG-H HEVC

<table>
<thead>
<tr>
<th>category</th>
<th>condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>none of the following</td>
</tr>
<tr>
<td>1</td>
<td>((c &lt; a) \land (c &lt; b))</td>
</tr>
<tr>
<td>2</td>
<td>(((c = a) \land (c &lt; b)) \lor ((c &lt; a) \land (c = b)))</td>
</tr>
<tr>
<td>3</td>
<td>(((c = a) \land (c &gt; b)) \lor ((c &gt; a) \land (c = b)))</td>
</tr>
<tr>
<td>4</td>
<td>((c &gt; a) \land (c &gt; b))</td>
</tr>
</tbody>
</table>

Edge offset mode

- Each sample is assigned to one of five categories
- Categorization depends on relation to neighboring samples in edge direction
- Transmit positive offset for four categories
- Smooth reconstructed signal in gradient direction
- Reduce ringing artifacts
Sample Adaptive Offset (SAO) in H.265 | MPEG-H HEVC

Second operation mode: Band offset mode
- Divide range of sample values into 32 equally sized bands
- Transmit offset for four consecutive bands (signal first band index)

Selection of operation mode on CTU level
- Three choices: Edge offset mode, band offset mode, no filtering
- Lagrangian mode decision can be applied
Coding experiments with H.265 | MPEG-H HEVC

- Two configurations: IPPP and IBBB coding
- Significant coding gains for IPPP coding
- Smaller coding gains for IBBB coding
- Support of bi-prediction already reduces ringing artifacts
Adaptive Wiener Filter

Signal-adaptive linear filter

- Apply linear filter to blocks of a picture or to the complete picture

\[
s'[x, y] = \sum_{k=-m}^{m} \sum_{\ell=-n}^{n} h_{k,\ell} \cdot s^*[x + k, y + \ell].
\]

- Encoder: Linear filter that minimizes distortion can be determined by solving linear equation system
- Filter coefficients have to be transmitted

Application

- Different variant: Separable or non-separable filter
- Not included in any video coding standards
- Considered as candidate for future standards
- H.264 | MPEG-4 AVC and H.265 | MPEG-H HEVC specify SEI message
  - Allows signaling of filter coefficients
  - Can be applied as non-normative post filter
Part Summary

Advanced motion-compensated prediction

- Multiple reference pictures: Block-adaptive selection
- Multi-hypothesis prediction: Typically adaptive bi-prediction
- Can be combined: Additive gains

Coding structures

- Simplest: Code pictures in display order
- Conventional B picture coding
- State-of-the-art: Hierarchical B pictures

In-Loop filters

- Deblocking filter
- Deringing filter / Sample adaptive offset
- Adaptive Wiener filter
- In-loop filters improve coding efficiency relative to post filters