ADVANCED VIDEO CODING

Fernando Pereira

Instituto Superior Técnico
There are two Parts in the MPEG-4 standard dealing with video coding:

- **Part 2: Visual (1998)** – Specifies several coding tools targeting the efficient and error resilient of video, including arbitrarily shaped video; it also includes coding of 3D faces and bodies.

- **Part 10: Advanced Video Coding (AVC) (2003)** – Specifies more efficient (about 50%) and more resilient frame based video coding tools; this Part has been jointly developed by ISO/IEC MPEG and ITU-T through the Joint Video Team (JVT) and it is often known as H.264/AVC.

Each of these two Parts specifies several profiles with different video coding functionalities and compression efficiency versus complexity trade-offs. H.264/AVC only addresses rectangular frames!
MPEG-4 Advanced Video Coding (AVC), also ITU-T H.264
H.264/AVC: The Objective

Coding of frame-based video with increased efficiency: about 50% less rate for the same quality regarding existing standards such as H.263, MPEG-2 Video and MPEG-4 Visual.

This standard (joint between ISO/IEC MPEG and ITU-T) offers also good flexibility in terms of efficiency-complexity trade-offs as well as good performance in terms of error resilience for mobile environments and fixed and wireless Internet (both progressive and interlaced formats).
H.264/AVC: Main Functionalities

- **Coding efficiency** *(about 50% compared to previous standards)*
  - Enhanced motion compensation, multiple reference frames, hierarchical B frames, improved de-blocking filter, smaller blocks for transform, enhanced entropy coding

- **Error resilience**
  - Data partitioning, FMO, slices, redundant slices, resynchronization markers, multiple reference pictures, parameter sets, etc.

- **Temporal scalability**
  - Enabled by flexible reference picture management via hierarchical predictions

- **Network friendliness**
  - Enabled through NAL design, bitrate adaptation
H.264/AVC: Applications

- **Entertainment Video (1-8+ Mbps, higher latency)**
  - Broadcast / Satellite / Cable / DVD / VoD / FS-VDSL / …
  - DVB/ATSC/SCTE, DVD Forum, DSL Forum

- **Conversational Services (usually <1 Mbps, low latency)**
  - H.320 Conversational
  - 3GPP Conversational H.324/M
  - H.323 Conversational Internet/best effort IP/RTP
  - 3GPP Conversational IP/RTP/SIP

- **Streaming Services (usually lower bitrate, higher latency)**
  - 3GPP Streaming IP/RTP/RTSP
  - Streaming IP/RTP/RTSP (without TCP fallback)

- **Other Services**
  - 3GPP Multimedia Messaging Services

H.264/AVC was adopted as recommended codec for all 3GPP video services.
As usual for MPEG standards, H.264/AVC specifies only the bitstream syntax and semantics as well as the decoding process:

- Allows several types of encoder optimizations
- Allows to reduce the encoding implementation complexity (at the cost of some quality)
- Does NOT allow to guarantee any minimum level of quality (which depends on the encoder implementations)!
To address the need for flexibility and customizability, the H.264/AVC design covers:

- A **Video Coding Layer (VCL)**, which is designed to efficiently represent the video content
- A **Network Abstraction Layer (NAL)**, which formats the VCL video representation and provides header information in a manner appropriate for conveyance by a variety of transport layers or storage media
The Network Abstraction Layer (NAL)
• **NAL Unit Definition** - The elementary unit processed by an H.264/AVC codec is called the network abstraction layer (NAL) unit. **The coded video data is organized into NAL units, which are packets containing an integer number of bytes.**

• **NAL Unit Encapsulation** - NAL units can be easily encapsulated into different transport protocols and file formats, such as MPEG-2 Systems transport stream, Real-Time Transfer Protocol (RTP), and MPEG-4 file format.

• **NAL Unit Composition** - A NAL unit starts with a one-byte header, which signals the type of contained data; the remaining bytes represent payload data. The header indicates the type of NAL unit and whether a VCL NAL unit is part of a reference or non-reference picture.

• **NAL Unit Classification:**
  - VCL NAL units - contain data representing the values and samples of video pictures in the form of slices or slice data partitions.
  - Non-VCL NAL units - contain associated additional information such as enhancement information (SEI), parameter sets, picture delimiter, or filler data.
• **Access Unit** - Set of consecutive NAL units with specific properties; the decoding of an access unit results in exactly one decoded picture.

• **Coded Video Sequence** - Set of consecutive access units with certain properties.

• **The most important non-VCL NAL units are parameter sets (PS) and Supplemental Enhancement Information (SEI).**
  - The parameter sets allow sending infrequently changing sequence (SPS) and picture (PPS) level information reliably, asynchronously, and in advance of the media stream that contains the VCL NAL units.
  - To be able to change picture parameters, such as picture size, without the need to transmit parameter set updates synchronously to the slice packet stream, the encoder and decoder can maintain a list of more than one SPS and PPS. Each slice header contains a codeword that indicates the SPS and PPS in use.
  - SEI messages are not required for decoding the samples of a video sequence; they rather provide additional information which can assist the decoding process or related processes like bitstream manipulation or display.
The Video Coding Layer (VCL)
H.264/AVC Basic Encoding Architecture: Again Predictive Coding
Similar to Previous Standards

- Macroblocks: 16×16 luma + 2 × 8×8 chroma samples (starting with 4:2:0 MBs)

- Input: Association of luma and chroma and conventional sub-sampling of chroma (starting with 4:2:0, more later)

- Block motion displacement

- Motion vectors over picture boundaries

- Variable block-size motion

- Block transforms

- Scalar quantization

- I, P, and B coding types
The H.264/AVC standard is based on the same predictive/hybrid coding architecture used for previous video coding standards with some important differences:

- Variable (and smaller) block size motion compensation
- Multiple reference frames
- Hierarchical transform with smaller block sizes
- Deblocking filter in the prediction loop
- Improved, adaptive entropy coding

which all together allow achieving substantial gains regarding the bitrate needed to reach a certain quality level.

The H.264/AVC standard addresses a vast set of applications, from personal communications to storage and broadcasting, at various qualities and resolutions.
Controlling Coding Performance

When using a standard with a **completely specified decoder**, parameters in the **non-specified encoder** should be selected such that good RD performance is achieved:

- **Global parameter selection** - Selects the appropriate temporal and spatial resolution of the video based on application, profile, and level constraints. Also, packetization modes, like slice sizes, are usually fixed for the entire session. The parameters are mainly determined by general application constraints.

- **Encoder control** - Performs local decisions, such as the selection of macroblock modes, reference frames, or motion vectors, on the macroblock level and below most appropriately based on a rate-distortion optimized mode selection applying Lagrangian techniques.

- **Rate control** - Controls the timing and bitrate constraints of the application by adjusting the Quantization Parameter or Lagrange parameter and is usually applied to achieve a constant bit rate (CBR) encoded video suitable for transmission over CBR channels.
Partitioning of the Picture

- **Picture (Y,Cr,Cb; 4:2:0 and later more; 8 bit/sample):**
  - A picture (frame or field) is split into 1 or several slices

- **Slice:**
  - Slices are self-contained
  - Slices are a sequence of macroblocks

- **Macroblock:**
  - Basic syntax & processing unit
  - Contains 16×16 luminance samples and 2×8×8 chrominance samples (4:2:0 content)
  - Macroblocks within a slice depend on each other
  - Macroblocks can be further partitioned
Slices and Slice Groups

Flexible Macroblock Ordering (FMO) allows mapping of MBs to slice groups, where a slice group itself may contain several slices. Therefore, MBs might be transmitted out of raster scan order in a flexible and efficient way.

• Slice Group:
  - Pattern of macroblocks defined by a Macroblock Allocation Map
  - A slice group may contain 1 to several slices

• Macroblock Allocation Map Types:
  - Interleaved slices
  - Dispersed macroblock allocation
  - Explicitly assign a slice group to each macroblock location in raster scan order
  - One or more “foreground” slice groups and a “leftover” slice group
Slice Coding Modes

• **Intra (I) Slices** - A slice in which all macroblocks are coded using intra coding:
  - **Intra versus IDR pictures** - There are IDR (instantaneous decoding refresh) pictures and regular intra-pictures whereby the latter do not necessarily provide the random access property as pictures before the intra pictures may be used as reference for succeeding predictively coded pictures.
  - **Constrained Intra** - The intra mode can be modified such that intra-prediction from predictively coded macroblocks is disallowed. The corresponding constraint intra flag is signaled in the PPS (picture parameter set).

• **P Slices** - In addition, some P slice macroblocks can also be coded using inter prediction with at most one motion-compensated prediction signal per prediction block (past or future).

• **B Slices** - In addition, some B slice macroblocks can also be coded using inter prediction with two motion-compensated prediction signals per prediction block (past or future).
The H.264/AVC Coding Tools
To increase Intra coding compression efficiency, it is possible to exploit for each MB the correlation with adjacent blocks or MBs in the same picture.

- If a block or MB is Intra coded, a prediction block or MB is built based on the previously coded and decoded blocks or MBs in the same picture.

- The prediction block or MB is subtracted from the block or MB currently being coded.

- To guarantee slice independency, only samples from the same slice can be used to form the Intra prediction.

This type of Intra coding may imply error propagation if the prediction uses adjacent MBs which have been Inter coded; this may be solved by using the so-called Constrained Intra Coding Mode where only adjacent Intra coded MBs are used to form the prediction.
Intra predictions may be performed in several ways:

1. **Luminance Intra16×16** - Single prediction for the whole MB: four modes are possible (vertical, horizontal, DC and planar) -> more appropriate uniform areas!

2. **Luminance Intra4×4** - Different predictions for the 16 samples of the several 4×4 blocks in a MB: nine modes (DC and 8 directional modes) -> more appropriate for textured areas!

3. **Chrominance** - Single prediction with four modes (vertical, horizontal, DC and planar)

- e.g., Mode 3: diagonal down/right prediction a, f, k, p are predicted by \((A + 2Q + l + 2) >> 2\)
16×16 Blocks Intra Prediction Modes

- The luminance is predicted in the same way for all samples of a 16×16 MB (Intra16×16 modes).

- This coding mode is adequate for the image areas which have a smooth variation.
4x4 Intra Blocks Intra Prediction Modes

Mode 0: Vertical
Mode 1: Horizontal
Mode 2: DC
Mode 3: Diag. Down-left
Mode 4: Diag. Down-right
Mode 5: Vertical-right
Mode 6: Horizontal-down
Mode 7: Vertical-left
Mode 8: Horizontal-up
Variable Block-Size Motion Compensation

Audiovisual Communication, Fernando Pereira, 2012/2013
Flexible Motion Compensation

- **Motion accuracy** - Each MB may be divided into several fixed size partitions used to describe the motion with \( \frac{1}{4} \) pel accuracy.

- **Partition types** - There are several partition types, from 4×4 to 16×16 luminance samples, with many options between the two limits.

- **Luminance partitions** - The luminance samples in a Inter coded MB (16×16) may be divided in four ways - Inter16×16, Inter16×8, Inter8×16 and Inter8×8 – corresponding to the four prediction modes at MB level.

- **P-slices partitions** - For P-slices, if the Inter8×8 mode is selected, each sub-MB (with 8×8 samples) may be divided again (or not), obtaining 8×8, 8×4, 4×8 and 4×4 partitions which correspond to the four predictions modes at sub-MB level.

For example, a maximum of 16 motion vectors may be used for a P coded (16×16) MB.
Motion vectors are differentially coded but not across slices.
Quarter-Sample Luma Interpolation

- Half sample positions are obtained by applying a 6-tap filter with tap values:
  
  \((1, -5, 20, 20, -5, 1)\)

- Quarter sample positions are obtained by averaging samples at integer and half sample positions

Audiovisual Communication, Fernando Pereira, 2012/2013
Multiple Reference Frames
The H.264/AVC standard supports motion compensation with multiple reference frames, meaning that more than one previously coded picture may be simultaneously used as prediction reference for the motion compensation of the MBs in a picture (at the cost of memory and computation).

- Both the encoder and the decoder store the reference frames in a memory with multiple frames; up to 16 reference frames are allowed.
- The decoder stores in the memory the same frames as the encoder; this is guaranteed by means of memory control commands which are included in the coded bitstream.
The Benefits of Multiple Reference Frames

H.264

Other Standards

H.264 can recognize periodic motion
The B frame concept is generalized in the H.264/AVC standard since now any frame may use as prediction reference for motion compensation also the B frames; this means the selection of the prediction frames only depends on the memory management performed by the encoder.

- B-slices use two reference frames, referred as the first and second reference frames, either from the past (list 0) or from the future (list 1).
- The selection of the two reference frames to use depends on the encoder.
- Blocks or MBs may be coded using a weighted prediction of two blocks or MBs in two reference frames, both in the past, both in the future, or one in the past and another in the future.
- The weighted prediction allows to reach a more efficient Inter coding this means with a lower prediction error.
New Types of Temporal Referencing

Known dependencies, e.g.
MPEG-1 Video, MPEG-2 Video, etc.

New types of dependencies:

• Referencing order and display order are decoupled, e.g. a P frame may not use as its prediction the previous P frame

• Referencing ability and picture type are decoupled, e.g. it is possible to use a B frame as reference
Hierarchical Prediction Structures

- Dyadic temporal scalability

- Low-delay prediction structure (structural delay is 0)
H.264/AVC B-Slices: Very Different from the Past

- H.264/AVC B-slices may serve as prediction for other slices.
- H.264/AVC B-slices MBs may use two references but now both in the past, both in the future or one in the past and another in the future.
- H.264/AVC B-slices don’t have to use any specific previous and next frames due to the availability of multiple reference frames.
- H.264/AVC B-slices may be configured to provide ‘low delay’ (using only references from the past).
- H.264/AVC B-slices are more complex than H.264/AVC P-slices, notably in terms of the memory bandwidth (double fetching).
Weighted Prediction for P and B Slices

• For each MB partition, it is possible to use a weighted prediction obtained from one or two reference frames.

• In addition to shifting in spatial position, and selecting from among multiple reference pictures, each region’s prediction sample values can be multiplied by a weight, and given an additive offset.

• For B-MBs, the weighted prediction may consist in performing motion compensation from the two reference frames and compute the prediction using a set weights $w_1$ and $w_2$.

• Some key uses: improved efficiency for B-slice coding, e.g., accelerating motion, illumination variations and representation of fades (fade-in, fade-out, and cross-fade from scene-to-scene).
Performance versus Prediction Structure

Video Coding Experiment with H.264/AVC
Foreman, CIF 30Hz @ 132 kbit/s

Performance as a function of $M$

Cascaded QP assignment
$\text{QP}(P) \approx \text{QP}(B_0) - 3 \approx \text{QP}(B_1) - 4 \approx \text{QP}(B_2) - 5$

- $M=1$
  - $I$ $P$ $P$ $P$ $P$ $P$ $P$ $P$

- $M=2$
  - $I$ $B_0$ $P$ $B_0$ $P$ $B_0$ $P$ $B_0$ $P$

- $M=4$
  - $I$ $B_1$ $B_0$ $B_1$ $P$ $B_1$ $B_0$ $B_1$ $P$

- $M=8$
  - $I$ $B_2$ $B_1$ $B_2$ $B_0$ $B_2$ $B_1$ $B_2$ $B_1$ $B_2$ $P$

Audiovisual Communication, Fernando Pereira, 2012/2013
Motion Estimation is King ...

- Variable block size (16x16 – 4x4) + quarter-pel + multi-frame motion compensation (H.264/AVC, 2003)
- Variable block size (16x16 – 8x8) (H.263, 1996) + quarter-pel motion compensation (MPEG-4, 1998)
- Frame Difference coding (H.120 1988)

Bit-rate Reduction: 75%

- Conditional Replenishment (H.120)
- Integer-pel motion compensation (H.261, 1991)
- Intraframe DCT coding (JPEG, 1990)

Foreman
10 Hz, QCIF
100 frames

PSNR [dB]

Rate [kbit/s]
Multiple Transforms

The H.264/AVC standard uses three transforms depending on the type of prediction residue to code:

1. 4×4 Hadamard Transform for the luminance DC coefficients in MBs coded with the Intra16×16 mode

2. 2×2 Hadamard Transform for the chrominance DC coefficients in any MB

3. 4×4 Integer Transform based on DCT for all the other blocks
Hierarchical Transform

Intra_16x16 macroblock type only: Luma 4x4 DC

Luma 4x4 block order for 4x4 Intra prediction and 4x4 residual coding

Chroma 4x4 block order for 4x4 residual coding, shown as 16-25, and Intra4x4 prediction, shown as 18-21 and 22-25

Integer DCT

Hadamard
The H.264/AVC standard uses an integer transform to code the prediction residue.

- The transform is applied to $4 \times 4$ blocks using a separable transform with properties similar to a $4 \times 4$ DCT

\[
C_{4 \times 4} = T_v \cdot B_{4 \times 4} \cdot T_h^T
\]

- $T_v, T_h$: vertical and horizontal transform matrixes

\[
T_v = T_h = \begin{bmatrix}
1 & 1 & 1 & 1 \\
2 & 1 & -1 & -2 \\
1 & -1 & -1 & 1 \\
1 & -2 & 2 & -1
\end{bmatrix}
\]

- $4 \times 4$ Integer DCT Transform
  - Easier to implement (only sums and shifts)
  - No mismatch in the inverse transform
The Hadamard transform has a set of basis functions including only ‘+1’ and ‘-1’ making its implementation rather simple (no multiplications are needed).

Although it may still have a frequencial interpretation like the DCT, as its basis functions have an increasing number of signal transitions, its energy compactation capability is typically smaller.
Quantization

- Quantization removes irrelevant information from the pictures to obtain a rather substantial bitrate reduction.

- **Quantization** corresponds to the division of each coefficient by a quantization factor while **inverse quantization** (reconstruction) corresponds to the multiplication of each coefficient by the same factor (there is always a quantization error involved ...).

- In H.264/AVC (first 3 profiles), **scalar quantization** is performed with the same quantization factor for all the transform coefficients in the MB; the FRExt profiles, allow sending a user-specific weighting quantization matrix.

- One out of 52 possible values for the quantization factor ($Q_{\text{step}}$) is selected for each MB indexed through the quantization step ($Q_p$) using a table which defines the relation between $Q_p$ and $Q_{\text{step}}$.

- The table above has been defined in order to have a reduction of about 12.5% in the bitrate for an increment of ‘1’ in the quantization step value, $Q_p$.

- The quantization step may be changed at MB level.
The Blocking Effect …
Blocking Effect: the Origin

There are two main building blocks within the H.264/AVC architecture which can be a source of blocking artifacts:

1. **IDCT with quantization** - The most significant factor is the block-based IDCT in intra and inter prediction error coding. Coarse quantization of the IDCT coefficients can cause visually disturbing discontinuities at the block boundaries.

2. **Motion compensated prediction** - Motion compensated blocks are generated by copying interpolated samples from different locations of possibly different reference frames. Since there is almost never a perfect fit for this data, discontinuities on the edges of the copied blocks of data typically arise. Additionally, in the copying process, existing edge discontinuities in reference frames are carried into the interior of the block to be compensated.

A deblocking filter is many times an advantageous tool to maximize the subjective quality and the compression performance.
Deblocking Filter Approaches

There are two main approaches to integrate deblocking filters into video codecs, notably as post-loop filters and in-loop filters:

- **POST-LOOP FILTERS** - Only operate on the display buffer outside of the coding loop, and, thus, are not normative in the standardization process. Because their use is optional, post-filters offer maximum freedom for decoder implementations.

- **IN-LOOP FILTERS** - Operate within the coding loop where the filtered frames are used as reference frames for motion compensation of subsequent coded frames. This forces all standard compliant decoders to perform identical filtering to stay in synchronization with the encoder. Naturally, a decoder can still perform post-filtering in addition to the in-loop filtering, if found necessary in a specific application.
  - Guarantee a certain level of quality
  - No need for extra frame buffer in the decoder
  - Improve objective and subjective quality with reduced decoding complexity
H.264/AVC Deblocking: Adaptive, In-Loop Approach

The H.264/AVC standard specifies the use of an adaptive, in-loop deblocking filter operating at the block edges with the target to increase the final subjective and objective qualities.

The filter performs simple operations to detect and analyze artifacts on coded block boundaries and attenuates those by applying a selected filter.

• This same filter needs to be present at the encoder and decoder (thus normative) since the filtered blocks are after used for motion estimation (in-loop filter). This filter has a performance superior to a post-processing filter (not in-loop and thus not normative).

• This filter has the following main advantages:
  - Blocks edges are smoothed without making the image blurred, improving the subjective quality.
  - The filtered blocks are used for motion compensation resulting in smaller residues after prediction, this means reducing the bitrate for the same target quality.
H.264/AVC Deblocking: Basics

- **Target** - In deblocking filtering, it is essential to be able to distinguish between *true* edges in the image and those created by quantization and motion compensation. To preserve image sharpness, the *true* edges should be left unfiltered as much as possible while filtering the *artificial* edges to reduce their visibility.

- **Basic idea** - A big difference between samples at the edges of two blocks should only be filtered if it can be attributed to quantization or motion compensation; otherwise, that difference must come from the image itself and, thus, should not be filtered.

- **Method** - The filter is applied to the vertical and horizontal edges of all 4×4 blocks in a MB; it is adaptive to the content, essentially removing the block effect without necessarily smoothing the image:
  - At slice level, the filter strength may be adjusted to the characteristics of the video sequence.
  - At the edge block level, the filter strength is adjusted depending on the type of coding (Intra or Inter), the motion and the coded residues.
  - At the sample level, sample values and quantizer-dependent thresholds can turn off filtering for each individual sample.
H.264/AVC Deblocking: Adaptability Control

- **Control parameter** - The adaptive filter is controlled through a *Boundary-Strength* (*Bs*) parameter which is allocated, at the decoder, to every edge between two 4×4 luminance sample blocks to define the filter strength. The value depends on the modes and coding conditions of the two adjacent (horizontal or vertical) blocks.

- **Parameter values** - A value of ‘4’ for *Bs* means that a special filter mode is applied, allowing for the strongest filtering, whereas a value of ‘0’ means no filtering is applied on this specific edge. In the standard mode of filtering, which is applied for edges with *Bs* from ‘1’ to ‘3’, the value of *Bs* determines the maximum modification of the sample values.

<table>
<thead>
<tr>
<th>Block modes and conditions</th>
<th>Bs</th>
</tr>
</thead>
<tbody>
<tr>
<td>One of the blocks is Intra <em>and</em> the edge is a macroblock edge</td>
<td>4</td>
</tr>
<tr>
<td>One of the blocks is Intra</td>
<td>3</td>
</tr>
<tr>
<td>One of the blocks has coded residuals</td>
<td>2</td>
</tr>
<tr>
<td>Difference of block motion ≥ 1 luma sample distance</td>
<td>1</td>
</tr>
<tr>
<td>Motion compensation from different reference frames</td>
<td>1</td>
</tr>
<tr>
<td>Else</td>
<td>0</td>
</tr>
</tbody>
</table>

- **Causes** - The gradation of *Bs* (in the table) reflects that the strongest blocking artifacts are mainly due to intra and prediction error coding and are to a smaller extent caused by block motion compensation.

- **Method** - Conditions are evaluated from top to bottom, until one of the conditions holds true, and the corresponding value is assigned to *Bs*. For *Bs* = 0, no sample is filtered while for *Bs* = 4 the filter reduces the most the block effect.
H.264/AVC Deblocking: Sample Level

- Up to three samples for luminance and one for chrominance on each side of the edge may be modified by the filtering process.

- Filtering on a line of samples only takes place if these three conditions all hold:
  \[
  \begin{align*}
  |p_0 - q_0| &< \alpha(\text{Index}_A) \\
  |p_1 - p_0| &< \beta(\text{Index}_B) \\
  |q_1 - q_0| &< \beta(\text{Index}_B)
  \end{align*}
  \]

- In these conditions, both table-derived thresholds are dependant on the average quantization parameter (QP) employed over the edge, as well as encoder selected offset values that can be used to control the properties of the deblocking filter at the slice level.

- The dependency of $\alpha$ and $\beta$ on QP links the strength of filtering to the general quality of the reconstructed picture prior to filtering.
H.264/AVC Deblocking: Subjective Result for Intra Coding at 0.28 bit/sample

1) Without filter  2) With H.264/AVC deblocking
H.264/AVC Deblocking: Subjective Result for Strong Inter Coding

1) Without Filter        2) With H.264/AVC deblocking
SOLUTION 1

• Exp-Golomb Codes are used for all symbols with the exception of the transform coefficients

• Context Adaptive VLCs (CAVLC) are used to code the transform coefficients
  - No end-of-block is used; the number of coefficients is decoded
  - Coefficients are scanned from the end to the beginning
  - Contexts depend on the coefficients themselves

SOLUTION 2 (5-15% less bitrate)

• Context-based Adaptive Binary Arithmetic Codes (CABAC)
  - Adaptive probability models are used for the majority of the symbols
  - The correlation between symbols is exploited through the creation of contexts
  - Restriction to binary arithmetic coding
Error Resilience Tools Summary

- Slice Structured Coding
- Flexible Macroblock Ordering (FMO)
- Arbitrary Slice Ordering (ASO)
- Slice Data Partitioning
- Intra Coding
- Redundant Slices
- Flexible Reference Frame Concept
- Switching Pictures
The H.264/AVC Video Symbolic Model

A video sequence is represented as a succession of pictures (interlaced or progressive) constituted by I, P and B slices, structured in macroblocks, each of them represented using motion vectors and/or quantized transform coefficients, following the constraints imposed by the slice coding type.
Error Robustness

• Typically, codec-level error resilience tools decrease compression efficiency. Ideally, compression and error robustness should be separately treated following Shannon’s separation principle. Practice is different …

• Since errors cannot be completely avoided, the following system design principles are relevant:
  - Error correction before the coding layer - Minimize the amount of losses in the wireless channel without completely sacrificing the video bit rate (e.g. FEC and retransmission).
  - Error detection - If errors are unavoidable, detect and localize erroneous video data.
  - Prioritization methods - If losses are unavoidable, at least minimize loss rates for very important data (e.g. data partitioning).
  - Error recovery and concealment - In case of losses, minimize the visual impact of losses on the actual distorted image (e.g. slice resynchronization, SEI messages by encoder).
  - Encoder-decoder mismatch avoidance - Limit or completely avoid encoder and decoder mismatches resulting in annoying error propagation (e.g. IDR refresh, flexible references).
Adding Complexity to Buy Quality

Complexity (memory and computation) typically increases 4× at the encoder and 3× at the decoder regarding MPEG-2 Video, Main profile.

Problematic aspects:

• Motion compensation with smaller block sizes (memory access)
• More complex (longer) filters for the ¼ pel motion compensation (memory access)
• Multiframe motion compensation (memory and computation)
• Many MB partitioning modes available (encoder computation)
• Intra prediction modes (computation)
• More complex entropy coding (computation)
H.264/AVC Profiles

Baseline
- I and P slice
- Motion-compensated prediction
- CAVLC
- In-loop deblocking
- Intra prediction
- FMO
- Data partitioning

Main
- B slice
- CABAC
- 8x8 transform
- Monochrome format
- MBAFF
- Weighted prediction
- Scaling matrices

High
- 8x8 spatial prediction
- 8-10b sample bit depth
- 4:2:2 chroma format
- High 4:2:2 Intra
- High 10 (FRExt)
- High 10 Intra

Extended
- SI and SP slice
- Field coding

High 4:4:4
- 4:4:4 chroma format
- 8-14b sample bit depth
- Predictive lossless

High 4:4:4 Predictive
- High 4:4:4 Intra
- CAVLC 4:4:4 Intra

Audiovisual Communication, Fernando Pereira, 2012/2013
# H.264/AVC Profiles: Technical Summary

<table>
<thead>
<tr>
<th>Feature</th>
<th>CBP</th>
<th>BP</th>
<th>XP</th>
<th>MP</th>
<th>HIP</th>
<th>Hi10P</th>
<th>Hi422P</th>
<th>Hi444PP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sample depths (bits)</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8 to 10</td>
<td>8 to 10</td>
<td>8 to 14</td>
</tr>
<tr>
<td>Flexible macroblock ordering (FMO)</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Arbitrary slice ordering (ASO)</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Redundant slices (RS)</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Data Partitioning</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>SI and SP slices</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>B slices</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Interlaced coding (PicAFF, MBAFF)</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>CABAC entropy coding</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>8×8 vs. 4×4 transform adaptivity</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Quantization scaling matrices</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Separate Cb and C, QP control</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Monochrome (4:0:0)</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Separate color plane coding</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Predictive lossless coding</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>
H.264/AVC: a Success Story …

- 3GPP (recommended in rel 6)
- 3GPP2 (optional for streaming service)
- ARIB (Japan mobile segment broadcast)
- ATSC (preliminary adoption for robust-mode back-up channel)
- Blu-ray Disc Association (mandatory for Video BD-ROM players)
- DLNA (optional in first version)
- DMB (Korea - mandatory)
- DVB (specified in TS 102 005 and one of two in TS 101 154)
- DVD Forum (mandatory for HD DVD players)
- IETF AVT (RTP payload spec approved as RFC 3984)
- ISMA (mandatory specified in near-final rel 2.0)
- SCTE (under consideration)
- US DoD MISB (US government preferred codec up to 1080p)
- (and, of course, MPEG and the ITU-T)
H.264/AVC Patent Licensing

- As with MPEG-2 Parts and MPEG-4 Part 2 among others, the vendors of H.264/AVC products and services are expected to pay patent licensing royalties for the patented technology that their products use.

- The primary source of licenses for patents applying to this standard is a private organization known as MPEG LA (which is not affiliated in any way with the MPEG standardization organization); MPEG LA also administers patent pools for MPEG-2 Part 1 Systems, MPEG-2 Part 2 Video, MPEG-4 Part 2 Video, and other technologies.
Decoder-Encoder Royalties

• Royalties to be paid by end product manufacturers for an encoder, a decoder or both ("unit") begin at US $0.20 per unit after the first 100,000 units each year. There are no royalties on the first 100,000 units each year. Above 5 million units per year, the royalty is US $0.10 per unit.

• The maximum royalty for these rights payable by an Enterprise (company and greater than 50% owned subsidiaries) is $3.5 million per year in 2005-2006, $4.25 million per year in 2007-08 and $5 million per year in 2009-10.

• In addition, in recognition of existing distribution channels, under certain circumstances an Enterprise selling decoders or encoders both (i) as end products under its own brand name to end users for use in personal computers and (ii) for incorporation under its brand name into personal computers sold to end users by other licensees, also may pay royalties on behalf of the other licensees for the decoder and encoder products incorporated in (ii) limited to $10.5 million per year in 2005-2006, $11 million per year in 2007-2008 and $11.5 million per year in 2009-2010.

• The initial term of the license is through December 31, 2010. To encourage early market adoption and start-up, the License will provide a grace period in which no royalties will be payable on decoders and encoders sold before January 1, 2005.
Participation Fees (1)

- **TITLE-BY-TITLE** – For AVC video (either on physical media or ordered and paid for on title-by-title basis, e.g., PPV, VOD, or digital download, where viewer determines titles to be viewed or number of viewable titles are otherwise limited), **there are no royalties up to 12 minutes in length.** For AVC video greater than 12 minutes in length, royalties are the lower of (a) 2% of the price paid to the licensee from licensee’s first arms length sale or (b) **$0.02 per title.** Categories of licensees include (i) replicators of physical media, and (ii) service/content providers (e.g., cable, satellite, video DSL, internet and mobile) of VOD, PPV and electronic downloads to end users.

- **SUBSCRIPTION** – For AVC video provided on a subscription basis (not ordered title-by-title), **no royalties are payable by a system (satellite, internet, local mobile or local cable franchise) consisting of 100,000 or fewer subscribers in a year.** For systems with greater than 100,000 AVC video subscribers, the annual participation fee is $25,000 per year up to 250,000 subscribers, $50,000 per year for greater than 250,000 AVC video subscribers up to 500,000 subscribers, $75,000 per year for greater than 500,000 AVC video subscribers up to 1,000,000 subscribers, and $100,000 per year for greater than 1,000,000 AVC video subscribers.
Participation Fees (2)

• Over-the-air free broadcast – There are no royalties for over-the-air free broadcast AVC video to markets of 100,000 or fewer households. For over-the-air free broadcast AVC video to markets of greater than 100,000 households, royalties are $10,000 per year per local market service (by a transmitter or transmitter simultaneously with repeaters, e.g., multiple transmitters serving one station).

• Internet broadcast (non-subscription, not title-by-title) – Since this market is still developing, no royalties will be payable for internet broadcast services (non-subscription, not title-by-title) during the initial term of the license (which runs through December 31, 2010) and then shall not exceed the over-the-air free broadcast TV encoding fee during the renewal term.

• The maximum royalty for Participation rights payable by an Enterprise (company and greater than 50% owned subsidiaries) is $3.5 million per year in 2006-2007, $4.25 million in 2008-09 and $5 million in 2010.

• As noted above, the initial term of the license is through December 31, 2010. To encourage early marketplace adoption and start-up, the License will provide for a grace period in which no Participation Fees will be payable for products or services sold before January 1, 2006.
The H.264/AVC Family ...

AVC, Advanced Video Coding

MVC, Multiview Video Coding

SVC, Scalable Video Coding
Scalable Video Coding (SVC)

An H.264/AVC Extension
An Heterogeneous World ...
Non-Scalable Coding ...

NON scalable stream

Decoding 1  Decoding 2  Decoding 3
Quality or SNR Scalable Coding …

Scalable stream

Decoding 1

Decoding 2

Decoding 3

Audiovisual Communication, Fernando Pereira, 2012/2013
Scalable Video Coding: Objectives

Scalability is a functionality regarding the decoding of parts of the coded bitstream, ideally

1. while achieving an RD performance at any supported spatial, temporal, or SNR resolution that is comparable to single-layer coding at that particular resolution, and
2. without significantly increasing the decoding complexity.
For each spatial resolution (except the lowest), the scalable stream asks for a bitrate overhead regarding the corresponding alternative non-scalable stream, although the total bitrate is lower than the total simulcasting bitrate.
The SVC standard enables the encoding of a high-quality/resolution video bitstream from which it is possible to easily extract one or more lower-quality/resolution sub-bitstreams that can themselves be decoded with a complexity and quality similar to that achieved using the existing non-scalable H.264/AVC design with the same rate.

- SVC should provide functionalities such as graceful degradation in lossy transmission environments as well as bitrate, format, and power adaptation; this should provide enhancements to transmission and storage applications.

- Previous video coding standards, e.g. MPEG-2 Video and MPEG-4 Visual, already defined scalable codecs that were not successful due the characteristics of traditional video transmission systems, the significant loss in coding efficiency as well as the large increase in decoder complexity in comparison with non-scalable solutions.
SVC Scalability Types

- **Temporal**: change of frame rate
  - 30 Hz
  - 15 Hz
  - 7.5 Hz

- **Spatial**: change of frame size
  - QCIF
  - CIF
  - TV
  - HDTV

- **Fidelity**: change of quality (a.k.a. SNR)
  - High rate
  - Low rate
Scalable Hierarchical Coding

Base layer

1st enhancement layer

2nd enhancement layer

3rd enhancement layer
SVC Friendly Applications

• **Robust Video Delivery**
  - Adaptive delivery over error-prone networks and to devices with varying capabilities
  - Combine with unequal error protection
  - Guaranteed base layer delivery
  - Internet/mobile transmission

• **Scalable Storage**
  - Scalable export of video content
  - Graceful expiration or deletion
  - Surveillance DVR’s and Home PVR’s

• **Enhancement Services**
  - Upgrade delivery from 1080i/720p to 1080p
  - DTV broadcasting, optical storage devices
SVC Alternatives

• **Simulcast**
  - Simplest solution
  - Code each layer as an independent stream
  - Incurs increase of rate

• **Stream Switching**
  - Viable for some application scenarios
  - Lacks flexibility within the network
  - Requires more storage/complexity at server

• **Transcoding** *(direct digital-to-digital data conversion of one encoding to another, e.g. with different rate, resolution, format)*
  - Low cost, designed for specific application needs
  - Already deployed in many application domains
Main SVC Requirements

- **Efficiency** - Similar **compression efficiency** compared to single-layer coding for each subset of the scalable bit stream.

- **Complexity** - Little increase in **decoding complexity** compared to single-layer decoding that scales with the decoded spatio-temporal resolution and bitrate.

- **Scalability Types** - Support of **temporal**, **spatial**, and **quality scalability**.

- **Compatibility** - Support of a **backward compatible base layer** (H.264/AVC in this case); this allows an H.264/AVC decoder to consume the base layer of SVC streams.

- **Adaptations** - Support of simple **bitstream adaptations** after encoding.
Similar to H.264/AVC, SVC organizes the encoded video data into network abstraction layer (NAL) units.
Temporal Scalability

General rule:
- Layers with higher temporal resolution cannot be used for prediction of layers with lower temporal resolution

Hierarchical prediction structures:
- Efficient method for enabling temporal scalability
- Not restricted to dyadic temporal scalability
- Can be combined with multiple reference pictures
- Delay can be controlled by restricting the use of future pictures as prediction references
Spatial Scalability: Typical Encoder

- Each video layer is coded by an H.264/AVC encoder.
- Since the input video for all the layers are from the same content, there is high correlation between the frames at the same temporal position in two neighboring layers.
- To remove such redundancy, there are inter-layer prediction (ILP) techniques which may be regarded as an additional coding tool to the temporal prediction (inter prediction between frames) and spatial prediction (intra prediction within one frame) used in single layer video coding.
The main goal of inter layer prediction is to use as much lower layer information as possible for improving the RD performance of the enhancement layers:

- **Switchable prediction (with upsampling)**
- **Prediction of intra macroblocks (as in MPEG-2, H.263, MPEG-4)**
- **No prediction of inter MBs (unlike MPEG-2, etc.)**
- **Prediction of partitioning and motion information (new in SVC)**
- **Prediction of residual data, bi-linear, blockwise (new in SVC)**
Inter-Layer Motion Prediction

- Inter-layer motion prediction saves bits by reusing (with upsampling if applicable) the motion data from the base layer; it works at MB partition level.

- Moreover, the MB type and partition information may also be derived from the base layer.

- The prediction data needed by the motion compensation process are completely derived from the base layer and only the residues are coded for the enhancement layer.
Inter-Layer Residual Prediction

- Inter-layer residual prediction was designed to reuse the residual signals from the base layer.

- When this prediction is enabled, the residues of the current layer are predicted (with unquantization, inverse transform, and upsampling if applicable) by those from the base layer and only the resulting differential signal is coded.

- The prediction is made block-wise (4×4 or 8×8 blocks) with a bi-linear upsampling filter.
**Inter-Layer Intra Prediction**

4-tap filter: [-3, 19, 19, -3]

- Inter-layer intra prediction implies the reconstruction of the base layer.
- Constrained intra prediction has to be used in the base layer to ensure that the inter-layer intra prediction will not reference to inter-frame predicted samples.

Audiovisual Communication, Fernando Pereira, 2012/2013
Spatial Scalability Performance

Spatial Scalability: Foreman, 150 pics, QCIF 15Hz @ 48 kbit/s -> CIF 15 Hz

- QCIF Base Layer
- CIF Single Layer
- QCIF & CIF Simulcast
- Overall Performance of SVC

Y-PSNR [dB] vs bit-rate [kbit/s]
Quality Scalability: Two Modes

- Quality scalable layers have the same spatio-temporal resolution but differ in fidelity.

- The H.264 SVC extension supports two quality scalable modes, with different scalability granularities, namely
  
  - Coarse Grain Scalability (CGS)
  
  - Medium Grain Scalability (MGS)
Typical Combined Scalability

- Layer indication is made by identifiers in the NAL unit header.
- Motion compensation and deblocking operations only at the target layer!
SVC Scalability Types: What Cost?

- **Temporal scalability** - Can be typically achieved without losses in rate-distortion performance.

- **Spatial scalability** - When applying an optimized SVC encoder control, the bitrate increase relative to non-scalable H.264/AVC coding, at the same fidelity, can be as low as 10% for dyadic (power of 2 related) spatial scalability. The results typically become worse as spatial resolution of both layers decreases and results improve as spatial resolution increases.

- **SNR scalability** - When applying an optimized encoder control, the bitrate increase relative to non-scalable H.264/AVC coding, at the same fidelity, can be as low as 10% for all supported rate points when spanning a bitrate range with a factor of 2-3 between the lowest and highest supported rate point.
While SVC supports up to 128 layers, the actual number of layers in an encoding depends on the application needs. With the currently specified profiles, the maximum number of enhancement layers is limited to 47 layers.
• Technically, the standard is a great success already with some adoption
  - *Google* Gmail service
  - *Vidyo* video conferencing for the Internet
  - Industry appears to be open towards embracing SVC for DTV broadcast services
  - Specifically, enhancement of 720p to 1080p

• Others might be less certain, but still possible …
  - SVC for surveillance recorders
Final Remarks on H.264/AVC and SVC Extension

• The H.264/AVC standard builds on previous coding standards to achieve a typical compression gain of about 50%, largely at the cost of increased encoder and decoder complexity.

• The compression gains are mainly related to the variable (and smaller) block size motion compensation, multiple reference frames, smaller blocks transform, deblocking filter in the prediction loop, and improved entropy coding.

• The H.264/AVC standard represents nowadays the state-of-the-art in video coding and it is currently being adopted by a growing number of organizations, companies and consortia.

• The SVC extension is technically powerful and their market relevance is already growing considering the increasing overall system heterogeneity.
What’s Next on Video Coding?
HDTV Displays are Everywhere …
Mobile Video is Exploding ...
More, More, More, More …

- Higher spatial resolutions
- Higher temporal resolutions
- From interlaced to progressive
- Higher pixel depths
- Higher number of views
- More colour, from 4:2:0 to 4:2:2 and 4:4:4
- …
- More content

Although content/cameras and displays seem to be ready for this ‘jumping up’ trend, the transmission infrastructure does not seem to be able to accommodate the associated growing rates!
The Next Step:
The HEVC Standard
Improved Video Compression: the Motivation

- Video is continually increasing by resolution
  - HD existing, UHD (4K×2K, 8K×4K) appearing
  - Mobile services going towards HD
  - Stereo, multi-view emerging

- Devices available to record and display ultra-high resolutions
  - Becoming affordable for home and mobile consumers

- Video has multiple dimensions to grow the data rate
  - Frame resolution, Temporal resolution
  - Color resolution, bit depth
  - Multi-view
  - Visible distortion still an issue with existing networks

- Necessary video data rate grows faster than feasible network transport capacities
  - Better video compression (than current H.264/AVC) needed in next decade

Audiovisual Communication, Fernando Pereira, 2012/2013
Main HEVC Requirements

• **Compression** - Substantially greater bitrate reduction over the H.264/AVC High profile is required for the target application(s); at no point of the entire bitrate range shall HEVC be worse than existing standard(s). Subjective visually lossless compression shall be supported.

• **Complexity** - Shall allow for feasible implementation within the constraints of the available technology at the expected time of usage. **HEVC should be capable of trading-off complexity and compression efficiency** by having: i) an operating point with significant decrease in complexity compared to H.264/AVC but with better compression efficiency than H.264/AVC; ii) an operating point with increase complexity and commensurate increase in compression performance.

• **Picture Formats** - Focus on a set of rectangular picture formats that will include all commonly used picture formats, ranging at least from VGA to 4K×2K, and potentially extending to QVGA and **8K×4K**.

• **Color Spaces and Color Sampling** - a) The YCbCr color space 4:2:0, 8 bits per component shall be supported; b) YCbCr/RGB **4:4:4** should be supported; c) Higher bit depth **up to 14 bits** per component should be supported.
Still using the same basic architecture, some more flexibility and adaptability (and thus complexity) may be added to the codec!

Growing the number of macroblock coding modes …
Video Compression Standard: a Toolbox Approach
What ‘Games’ Can we Play?

- Intra prediction
- Inter prediction
- Motion models
- Transform
- Quantization
- Entropy coding
- In-loop filtering
- ...

Audiovisual Communication, Fernando Pereira, 2012/2013
HEVC: Foreseen Applications

- Home cinema
- Digital Cinema
- Surveillance
- Broadcast
- Real-time communications, video chat, video conferencing
- Mobile streaming, broadcast and communications
- Storage
- Camcorders
- Video On Demand
- Internet streaming, download and play
- 3D video, telepresence
- Content production and distribution
- Medical imaging
HEVC: Achievements

• High Efficiency Video Coding (HEVC) is a video compression standard, a successor to H.264/AVC jointly developed by ISO/IEC Moving Picture Experts Group (MPEG) and ITU-T Video Coding Experts Group (VCEG) as MPEG-H Part 2 and ITU-T H.HEVC (corresponding to H.265).

• MPEG and VCEG have established a Joint Collaborative Team on Video Coding (JCT-VC) to develop the HEVC standard.

• HEVC improves the video compression by doubling the data compression factor compared to H.264/AVC for the same quality (50% of the rate for the same quality), and can support 8K UHD and resolutions up to (8192 × 4320).
For more details …

Return next year!
Final Conclusions
Video Coding: Overall Summary

- Video is a lively area of research, major progress in standardization during last decade, still ongoing

- H.264/AVC predominant in market currently, has taken over from H.262/MPEG-2 Video (Part 2)

- Developments in multi-view, 3D still ongoing

- HEVC shows significant technical and performance advance over prior standards
  - Roughly 50% rate reduction over H.264/AVC, more over other standards
  - Specific support for parallelism to meet today's implementation challenges
  - Only one "Main" Profile for various applications (mobile, high-quality entertainment, etc.) in first version preferable
  - Future extensions planned, notably for scalable and 3D video coding
Bibliography

• **The MPEG-4 Book**, F. Pereira, T. Ebrahimi, Prentice Hall, 2002

• **H.264 and MPEG-4 Video Compression**, I. Richardson, John Wiley & Sons, 2003