Video Coding Standards
- An Overview

King Ngi Ngan

Department of Electronic Engineering
The Chinese University of Hong Kong
Outline

- Introduction
- Standards development
- Video coding principles
- Overview of video coding standards
  - H.261, MPEG-1, MPEG-2, H.263, MPEG-4, H.264
Introduction

• Why have standards?
  - To ensure interoperability amongst different manufacturers and to encourage interworking, competition and increased choice.

• What is a video coding standard?
  - It defines a coded representation (or syntax) that describes visual data in a compressed form, and a method of decoding the syntax to reconstruct the visual information.
  - It aims to ensure that compliant encoders and decoders can successfully interwork with each other, whilst allowing manufacturers the freedom to develop competitive and innovative products.
Introduction

- The standards do not define the encoder; rather they define the output that an encoder should produce.
- Each standard defines a decoding method but the manufacturers are free to develop alternative decoders as long as they can decode the syntax, and produce the same result as that in the standard.
Standards Development

• Requirements
  - Identify the industry needs and requirements of the new standard
  - Determine the rules of the game for the competitive phase

• Competition
  - The standard and the technology behind it must be the state of the art.
  - It must bring together the best of academic and industrial research.
  - Extensive testing and evaluation must be carried out, so that new ideas are considered solely on the basis of their technical merits and the trade-off between quality and cost of implementation.
Standards Development

- Convergence
  - The ideas and techniques considered as promising at the end of the competitive phase are to be integrated into one solution.
  - A software model (*Reference Software*) is developed that implements the agreed functionality, together with a document describing this software (*Test Model*).
  - After several updates and revisions, the model document is converted into a Draft Standard.
  - Further refinements and a series of ballots by the National Bodies, the Draft Standard is elevated to *International Standard*. 
Video Coding Standards

- ITU-T Recommendation H.261
  - Video codec for audiovisual services at p×64 Kbits/s

- ISO/IEC 11172 (MPEG-1)
  - Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbits/s

- ITU-T Recommendation H.262 or ISO/IEC 13818 (MPEG-2)
  - Generic coding of moving pictures and associated audio information – Part 2: Video
Video Coding Standards

- ITU-T Recommendation H.263
  - Video coding for low bit rate communication

- ISO/IEC 14496 (MPEG-4)
  - Coding of audiovisual objects – Part 2: Visual

- ITU-T Recommendation H.264 or ISO/IEC 14496 (MPEG-4)
  - Part 10: Advanced video coding

- AVS – Audio Video coding Standard
  - China’s national standard

- AVS-M
  - Audio video coding standard for mobile phones?
Related Standards

• ISO/IEC 15938 (MPEG-7)
  – Multimedia content description interface

• ISO/IEC 18034 (MPEG-21)
  – Multimedia framework
Video Coding Standards Timeline

ITU-T Standards
- H.261
- H.263
- H.263+
- H.263++

Joint ITU-T/MPEG Standards
- H.262/MPEG-2
- H.263L/H.264/MPEG-4 Pt. 10

MPEG Standards
- MPEG-1
- MPEG-4

Timeline:
- 1984
- 1986
- 1988
- 1990
- 1992
- 1994
- 1996
- 1998
- 2000
- 2002
- 2004

5 June 2004
Video Coding Principles

Image blocks

Encoder

Bitstream

DCT

Q

VLC

VB

CC

Q⁻¹

DCT⁻¹

MC

FS

LF

5 June 2004
Prediction

- Video frames have high temporal correlation.
- The information to be coded can be significantly reduced by coding the residual image.

\[ \text{Frame } n - \text{Frame (} n - 1) = \text{Residual image} \]
Prediction

• Prediction can be forward, backward or bidirectional.
Motion compensation is to find the closest estimate of the data in the current frame from those in the previous frame.

Motion estimation is performed by block matching.
Motion Estimation

Frame \((n-1)\)

\[ M = \sum_{j=1}^{N} \sum_{k=1}^{N} W(j,k) \]

\[ MAD = \frac{1}{N^2} \sum_{j=1}^{N} \sum_{k=1}^{N} |W(j,k) - X(j,k)| \]
• Calculate the MAD for all positions of $X(j,k)$ within the search window $W(j,k)$.

• Find the minimum MAD to be the best match.

• Motion vector is defined by $MV(v_x,v_y)$. 
Motion Estimation

Frame \((n-1)\)

Frame \(n\)

Motion vectors
Discrete Cosine Transform (DCT)

- DCT decorrelates the image by transforming the pixels to frequency domain.
DCT

1 top-left coefficient

3 top-left coefficients

6 top-left coefficients

All coefficients

5 June 2004
Quantization

Quantizer with deadzone (Nonintra M-blocks)

Quantizer with no deadzone (Intra M-blocks)
Variable Length Coding (VLC)

<table>
<thead>
<tr>
<th>Level</th>
<th>VLC</th>
<th>Level</th>
<th>VLC</th>
<th>Level</th>
<th>VLC</th>
</tr>
</thead>
<tbody>
<tr>
<td>60</td>
<td>111</td>
<td>5</td>
<td>0010 111</td>
<td>51</td>
<td>0001 0010</td>
</tr>
<tr>
<td>4</td>
<td>1101</td>
<td>9</td>
<td>0010 110</td>
<td>23</td>
<td>0001 0001</td>
</tr>
<tr>
<td>8</td>
<td>1100</td>
<td>17</td>
<td>0010 101</td>
<td>43</td>
<td>0001 0000</td>
</tr>
<tr>
<td>16</td>
<td>1011</td>
<td>33</td>
<td>0010 100</td>
<td>25</td>
<td>0000 1111</td>
</tr>
<tr>
<td>32</td>
<td>1010</td>
<td>6</td>
<td>0010 011</td>
<td>37</td>
<td>0000 1110</td>
</tr>
<tr>
<td>12</td>
<td>1001 1</td>
<td>10</td>
<td>0010 010</td>
<td>26</td>
<td>0000 1101</td>
</tr>
<tr>
<td>48</td>
<td>1001 0</td>
<td>18</td>
<td>0010 001</td>
<td>38</td>
<td>0000 1100</td>
</tr>
<tr>
<td>20</td>
<td>1000 1</td>
<td>34</td>
<td>0010 000</td>
<td>29</td>
<td>0000 1011</td>
</tr>
<tr>
<td>40</td>
<td>1000 0</td>
<td>7</td>
<td>0001 1111</td>
<td>45</td>
<td>0000 1010</td>
</tr>
<tr>
<td>28</td>
<td>0111 1</td>
<td>11</td>
<td>0001 1110</td>
<td>53</td>
<td>0000 1001</td>
</tr>
<tr>
<td>44</td>
<td>0111 0</td>
<td>19</td>
<td>0001 1101</td>
<td>57</td>
<td>0000 1000</td>
</tr>
<tr>
<td>52</td>
<td>0110 1</td>
<td>35</td>
<td>0001 1100</td>
<td>30</td>
<td>0000 0111</td>
</tr>
<tr>
<td>56</td>
<td>0110 0</td>
<td>13</td>
<td>0001 1011</td>
<td>46</td>
<td>0000 0110</td>
</tr>
<tr>
<td>1</td>
<td>0101 1</td>
<td>49</td>
<td>0001 1010</td>
<td>54</td>
<td>0000 0101</td>
</tr>
<tr>
<td>61</td>
<td>0101 0</td>
<td>21</td>
<td>0001 1001</td>
<td>58</td>
<td>0000 0100</td>
</tr>
<tr>
<td>2</td>
<td>0100 1</td>
<td>41</td>
<td>0001 1000</td>
<td>31</td>
<td>0000 0011 1</td>
</tr>
<tr>
<td>62</td>
<td>0100 0</td>
<td>14</td>
<td>0001 0111</td>
<td>47</td>
<td>0000 0011 0</td>
</tr>
<tr>
<td>24</td>
<td>0011 11</td>
<td>50</td>
<td>0001 0110</td>
<td>55</td>
<td>0000 0010 1</td>
</tr>
<tr>
<td>36</td>
<td>0011 10</td>
<td>22</td>
<td>0001 0101</td>
<td>59</td>
<td>0000 0010 0</td>
</tr>
<tr>
<td>3</td>
<td>0011 01</td>
<td>42</td>
<td>0001 0100</td>
<td>27</td>
<td>0000 0001 1</td>
</tr>
<tr>
<td>63</td>
<td>0011 00</td>
<td>15</td>
<td>0001 0011</td>
<td>39</td>
<td>0000 0001 0</td>
</tr>
</tbody>
</table>
MPEG-2

Part 1: Systems

Part 2: Video

Part 3: Audio

Part 4: Conformance

Part 5: Software

Part 6: Digital Storage Media – Command and Control (DSM-CC)

Part 7: Non-Backward Compatible (NBC) Audio

Part 8: 10-Bit Video (Dropped)

Part 9: Real Time Interface

Part 10: DSM-CC Conformance
Features of MPEG-2

• Low delay
  – Can be used for real time applications

• Scalable coding
  – Provides spatial resampling and error resilience

• Format flexibility
  – Flexibility of frame size and frame rate, including interlaced format

• Generic
  – Supports a wide range of applications

• Random access
  – A coded bitstream is accessible in its middle and any frame is decodable in a limited amount of time.
Applications of MPEG-2

- Satellite TV broadcast service
- Cable TV
- Digital terrestrial TV broadcast
- Electronic cinema
- Electronic news gathering
- Home theatre
- Video conferencing
- Video service hierarchies with multiple spatial, temporal and quality resolutions
- HDTV with embedded TV
- Interactive and serial storage media (e.g. optical disks, digital VTR)
Profiles and Levels

- In order to maximize the interoperability while limiting the complexity, targeting the largest deployment of the standard, profiles and levels are specified.

- **Profile**: a subset of the entire bit stream syntax

- **Level**: a specified set of constraints imposed on values of the syntax elements in the bit stream
### Profiles and Levels

<table>
<thead>
<tr>
<th>Level</th>
<th>Spatial Resolution Layer</th>
<th>Profile</th>
<th>Simple</th>
<th>Main</th>
<th>SNR</th>
<th>Spatial</th>
<th>High</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>1920</td>
<td>1152</td>
<td>60</td>
<td>1920</td>
<td>1152</td>
</tr>
<tr>
<td>High</td>
<td>Enhancement</td>
<td>samples/line</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>lines/frame</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>frames/sec</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Low</td>
<td></td>
<td>samples/line</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>lines/frame</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>frames/sec</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>High-1440</td>
<td>Enhancement</td>
<td>samples/line</td>
<td>1440</td>
<td>1152</td>
<td>60</td>
<td>1440</td>
<td>1152</td>
</tr>
<tr>
<td></td>
<td></td>
<td>lines/frame</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>frames/sec</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lower</td>
<td></td>
<td>samples/line</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>lines/frame</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>frames/sec</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Main</td>
<td>Enhancement</td>
<td>samples/line</td>
<td>720</td>
<td>576</td>
<td>30</td>
<td>720</td>
<td>576</td>
</tr>
<tr>
<td></td>
<td></td>
<td>lines/frame</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>frames/sec</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lower</td>
<td></td>
<td>samples/line</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>lines/frame</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>frames/sec</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Low</td>
<td>Enhancement</td>
<td>samples/line</td>
<td>352</td>
<td>288</td>
<td>30</td>
<td>352</td>
<td>288</td>
</tr>
<tr>
<td></td>
<td></td>
<td>lines/frame</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>frames/sec</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lower</td>
<td></td>
<td>samples/line</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>lines/frame</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>frames/sec</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**NOTE** - In the case of single layer or SNR scaled coding, the limits specified by ‘Enhancement layer’ apply.
MPEG-2 Video Structure

- Source formats - 4:2:0, 4:2:2, and 4:4:4 format
MPEG-2 Video Coding Tools

Frame DCT coding

Field DCT coding
MPEG-2 Video Coding Tools

- For interlace in Frame-pictures, two features are provided:
  - Alternate scan to allow the more significant bottom-left coefficients to be sent first
  - Field DCT coding to increase the vertical correlation within the luminance blocks

- I, P and B pictures

- Four prediction modes:
  - Field prediction
  - Frame prediction
  - $16 \times 8$ motion compensation
  - Dual-prime prediction (for P pictures only)
    - Predictions are made from two reference fields and are averaged to form the final prediction
MPEG-2 Scalable Coding Tools

• Spatial Scalability
  - Involves generating two spatial resolution video layers from a single video source such that the lower layer is coded by itself to provide the basic spatial resolution and the enhancement layer employs the spatially interpolated lower layer and carries the full spatial resolution of the input video source.

• SNR Scalability
  - Involves generating two video layers of same spatial resolution but different video qualities from a single video source such that the lower layer is coded by itself to provide the basic video quality and the enhancement layer is coded to enhance the lower layer.
MPEG-2 Scalable Coding Tools

• Temporal Scalability
  - Involves partitioning of video frames into layers, whereas the lower layer is coded by itself to provide the basic temporal rate and the enhancement layer is coded with temporal prediction with respect to the lower layer, these layers when decoded and temporal multiplexed to yield full temporal resolution of the video source.

• Data Partitioning
  - The bitstream is partitioned between two channels such that more critical parts of the bitstream (such as headers, motion vectors, low frequency DCT coefficients) are transmitted in the channel with the better error protection, and less critical data (such as higher frequency DCT coefficients) is transmitted in the channel with poorer error protection to provide error resilience to the bitstream.
MPEG-4

Part 1: Systems
Part 2: Visual
Part 3: Audio
Part 4: Conformance Testing
Part 5: Reference Software
Part 6: Delivery Multimedia Integration Framework
Part 7: Optimized Visual Reference Software
Part 8: Carriage of MPEG-4 over IP
Part 9: Reference Hardware Description

Part 10: Advanced Video Coding

Part 11: Scene Description and Application Engine

Part 12: ISO Base Media File Format


Part 14: MPEG-4 File Format

Part 15: AVC File Format

Part 16: Animation Framework Extension
Features of MPEG-4 Part 2: Visual

- Efficient compression of progressive and interlaced natural (rectangular shape) video sequences. The core compression tools are based on the ITU-T H.263 standard.
- Coding of arbitrarily-shaped video objects
- Support for effective transmission over practical networks using error resilient and scalable coding tools
- Coding of still texture (image data)
- Coding of animated visual objects (2D & 3D polygonal meshes, animated face and human body)
- Coding of “studio” quality video
Applications of MPEG-4 Part 2: 
Visual

• “Legacy” video applications, e.g. digital TV broadcasting, videoconferencing and video storage

• “Object-based” video applications in which a video scene may be composed of a number of distinct video objects, each independently coded

• Rendered computer graphics using 2D and 3D deformable mesh geometry and/or animated human faces and bodies

• “Hybrid” video applications combining real-world (natural) video, still images and computer-generated graphics

• Streaming video over the Internet and mobile channels

• High-quality video editing and distribution for the studio production environment
Video Objects

- MPEG-4 treats a video sequence as a collection of video objects.

- A video object (VO) is an area of video scene that may occupy an arbitrary-shaped region and may exist for an arbitrary length of time.

- An instance of a VO at a particular point in time is a video object plane (VOP).

- In the traditional video coding sense, a rectangular video frame is a VOP and a video sequence is a VO.
Examples of Video Objects
### MPEG-4 Part 2: Visual

**For coding natural video:**

<table>
<thead>
<tr>
<th>MPEG-4 Visual Profile</th>
<th>Main Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simple</td>
<td>Low-complexity coding of rectangular video frames</td>
</tr>
<tr>
<td>Advanced Simple</td>
<td>Coding rectangular frames with improved efficiency and support for interlaced video</td>
</tr>
<tr>
<td>Advanced Real-Time Simple</td>
<td>Coding rectangular frames for real-time streaming</td>
</tr>
<tr>
<td>Core</td>
<td>Basic coding of arbitrary-shaped video objects</td>
</tr>
<tr>
<td>Main</td>
<td>Feature-rich coding of video objects</td>
</tr>
<tr>
<td>Advanced Coding Efficiency</td>
<td>Highly efficient coding of video objects</td>
</tr>
<tr>
<td>N-Bit</td>
<td>Coding of video objects with sample resolutions other than 8 bits</td>
</tr>
</tbody>
</table>
## MPEG-4 Part 2: Visual

### MPEG-4 Visual Profile | Main Features
--- | ---
Simple Scalable | Scalable coding of rectangular video frames
Fine Granular Scalable | Advanced scalable coding of rectangular frames
Core Scalable | Scalable coding of video objects
Scalable Texture | Scalable coding of still texture
Advanced Scalable Texture | Scalable coding of still texture with improved efficiency and object-based features
Advanced Core | Combines features of Simple, Core and Advanced Scalable Texture Profiles
Simple Studio | Object-based coding of high quality video sequences
Core Studio | Object-based coding of high quality video with improved compression efficiency
MPEG-4 Part 2: Visual

For coding synthetic or hybrid video:

<table>
<thead>
<tr>
<th>MPEG-4 Visual Profile</th>
<th>Main Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basic Animated Texture</td>
<td>2D mesh coding with still texture</td>
</tr>
<tr>
<td>Simple Face Animation</td>
<td>Animated human face models</td>
</tr>
<tr>
<td>Simple Face and Body Animation</td>
<td>Animated face and body models</td>
</tr>
<tr>
<td>Hybrid</td>
<td>Combines features of Simple, Core, Basic Animated Texture and Simple Face Animation</td>
</tr>
</tbody>
</table>

Each profile has several levels, depending on size, bit rate and number of objects.
Coding of rectangular frames

Input/Output formats:
4:2:0, 4:2:2 or 4:4:4 progressive or interlaced

<table>
<thead>
<tr>
<th>Profile</th>
<th>Coding Tools</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simple (VLBV Core – based on H.263)</td>
<td>I-VOP, P-VOP, 4MV, UMV, Intra prediction</td>
</tr>
<tr>
<td></td>
<td>Video packets, Data Partitioning, RVLCs</td>
</tr>
<tr>
<td></td>
<td>Short Header (compatible with H.263 Baseline)</td>
</tr>
<tr>
<td>Advanced Simple</td>
<td>Simple coding tools</td>
</tr>
<tr>
<td></td>
<td>B-VOP, Interlace, Alternate quantizer, Global MC,</td>
</tr>
<tr>
<td></td>
<td>Quarter pixel ME</td>
</tr>
<tr>
<td>Advanced Real Time Simple</td>
<td>Simple coding tools</td>
</tr>
<tr>
<td></td>
<td>Dynamic Resolution Conversion, NEWPRED</td>
</tr>
</tbody>
</table>
## Coding of arbitrary-shaped regions

<table>
<thead>
<tr>
<th>Profile</th>
<th>Coding Tools</th>
</tr>
</thead>
<tbody>
<tr>
<td>Core</td>
<td>Simple coding tools</td>
</tr>
<tr>
<td></td>
<td>B-VOP, Alternate quantizer, Binary shape, P-VOP</td>
</tr>
<tr>
<td></td>
<td>temporal scalability</td>
</tr>
<tr>
<td>Main</td>
<td>Core coding tools</td>
</tr>
<tr>
<td></td>
<td>Grey shape, Interlace, Sprite</td>
</tr>
<tr>
<td>Advanced Coding Efficiency</td>
<td>Core coding tools</td>
</tr>
<tr>
<td></td>
<td>Quarter pixel ME, Shape adaptive DCT, Global MC, Grey shape, Interlace</td>
</tr>
<tr>
<td>N-bit</td>
<td>Core coding tools</td>
</tr>
<tr>
<td></td>
<td>N-bit</td>
</tr>
</tbody>
</table>
## Scalable Video Coding

<table>
<thead>
<tr>
<th>Profile</th>
<th>Coding Tools</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simple Scalable</td>
<td>I-VOP, P-VOP, 4MV, UMV, Intra prediction Video packets, Data Partitioning, RVLCs B-VOP, Temporal scalability (rectangular), Spatial scalability (rectangular)</td>
</tr>
<tr>
<td>Fine Granular Scalable</td>
<td>Simple coding tools FGS, FGS Temporal scalability, Interlace, B-VOP, Alternate quantizer</td>
</tr>
<tr>
<td>Core Scalable</td>
<td>Core coding tools Temporal scalability (rectangular), Spatial scalability (rectangular), Object-based spatial scalability</td>
</tr>
</tbody>
</table>
Texture Coding

<table>
<thead>
<tr>
<th>Profile</th>
<th>Coding Tools</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scalable Texture</td>
<td>Scalable still texture (DWT)</td>
</tr>
<tr>
<td>Advanced Scalable Texture</td>
<td>Scalable Texture coding tools, Scalable shape coding, Texture error resilience, Wavelet tiling</td>
</tr>
</tbody>
</table>

Profile:

- Scalable Texture
- Advanced Scalable Texture

Coding Tools:

- Scalable still texture (DWT)
- Scalable Texture coding tools
- Scalable shape coding
- Texture error resilience
- Wavelet tiling

Diagram:

- DWT
- Quantizer
- Predictive Coding
- Arithmetic Encoder
- Coded bitstream

Still texture

DC subband

AC subband

Quantizer & Scanning

Zero-tree Coding

5 June 2004
### Studio Coding

<table>
<thead>
<tr>
<th>Profile</th>
<th>Coding Tools</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simple Studio</td>
<td>I-VOP, Studio slice, Studio DPCM, Studio binary and grey shape, Interlace,</td>
</tr>
<tr>
<td></td>
<td>Frame/Field</td>
</tr>
<tr>
<td>Core Studio</td>
<td>Simple Studio coding tools</td>
</tr>
<tr>
<td></td>
<td>P-VOP, Studio Sprite</td>
</tr>
</tbody>
</table>
• Coding is based on macroblock structure consisting of $16 \times 16$ Y block, $8 \times 8$ Cr and Cb blocks.

• A slice is a group of macroblocks

..... etc.
H.264 Coding Tools

• Intra prediction:
  - 13 modes for Y components (9 for INTRA_4×4 blocks and 4 for INTRA_16×16 blocks)
  - 4 modes for Cr & Cb components
  - Can use decoded macroblocks of the same image for prediction

• Motion compensated prediction:
  - Block sizes of 16×16, 16×8, 8×16 & 8×8
  - Sub-block sizes of 8×8 can be further divided into 8×4, 4×8 & 4×4

• Multiple reference frames (MRF)

• ¼ pixel motion estimation
H.264 Coding Tools

- Prediction mode is chosen by minimizing the rate-distortion function using Lagrangian optimization techniques.
- 4×4 integer DCT, and 4×4 or 2×2 HT
- Entropy coding:
  - Context-adaptive variable length codes (CAVLC)
  - Context-based binary arithmetic coding (CABAC)
- Adaptive deblocking filter
H.264 Profiles

- 3 profiles with 15 levels for each profile
  - Bit rates from 64 Kbps to 240 Mbps

Core Coding Tools:

- I & P slices, different block sizes, ¼ pixel MC, MRF, deblocking filter, CAVLC, Intra prediction
H.264 Profiles

- Typical applications:
  - Baseline
    - Video conferencing
    - Low-cost/low-complexity applications
  - Main
    - Broadcasting
    - HD-DVD
  - Extended
    - Video streaming
## Differences between MPEG-4 Visual & H.264

<table>
<thead>
<tr>
<th>Comparison</th>
<th>MPEG-4 Visual</th>
<th>H.264</th>
</tr>
</thead>
<tbody>
<tr>
<td>Supported data types</td>
<td>Rectangular video frames and fields, arbitrary-shaped video objects, still texture or synthetic-natural hybrid video objects, 2D &amp; 3D mesh objects</td>
<td>Rectangular video frames and fields</td>
</tr>
<tr>
<td>Number of profiles</td>
<td>19</td>
<td>3</td>
</tr>
<tr>
<td>Compression efficiency</td>
<td>Medium</td>
<td>High</td>
</tr>
<tr>
<td>Support for video streaming</td>
<td>Scalable coding</td>
<td>Switching slices</td>
</tr>
<tr>
<td>MC minimum block size</td>
<td>8×8</td>
<td>4×4</td>
</tr>
<tr>
<td>MV accuracy</td>
<td>Half or quarter-pixel</td>
<td>Quarter-pixel</td>
</tr>
<tr>
<td>Transform</td>
<td>8×8 DCT</td>
<td>4×4 ICT</td>
</tr>
<tr>
<td>Deblocking filter</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Licence payments required for</td>
<td>Yes</td>
<td>Yes (lower than MPEG-4)</td>
</tr>
<tr>
<td>commercial implementation</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

5 June 2004
Comparisons

- For video streaming applications
Comparisons

- For videoconferencing applications
Comparisons

- For entertainment-quality applications
References

• http://www.vcodex.com

• http://bs.hhi.de/mpeg-video


Thank you for your attention!