HEVC
The Next Generation Video Coding
Outline

• Introduction

• Technical Details
  • Coding structures
  • Intra prediction
  • Inter prediction
  • Transform
  • Scanning and Entropy Coding
  • In-loop filters
  • Parallel processing tools
What is HEVC?

- HEVC: High Efficiency Video Coding
- Wikipedia:
  - High Efficiency Video Coding (HEVC) is a video compression standard, a successor to H.264/MPEG-4 AVC (Advanced Video Coding), that was jointly developed by the ISO/IEC Moving Picture Experts Group (MPEG) and ITU-T Video Coding Experts Group (VCEG) as ISO/IEC 23008-2 MPEG-H Part 2 and ITU-T H.265.
Timeline

- 2010.01: Formal joint CfP from VCEG and MPEG
- 2010.04: JCT-VC team, HEVC joint project, full proposals
- 2010.07: TMuC SW ready, tool experiments (TE)
- 2010.10: HM SW ready, core experiments (CE)
- 2011.02: Work Draft (WD)
- 2012.02: Committee Draft (CD)
- 2013.04: First Approved Version (containing Main, Main 10, and Main Still Picture profiles)
- 2014.10: Second Approved Version (adding range extensions profiles)
Timeline

- The specification was formally ratified as a standard on April 13, 2013.
HEVC Involved Companies
Goals

- Achieve a compression gain of 50% over H.264/AVC
- $x_{10}$ complexity max for encoder and $x_2/x_3$ max for decoder
- Encoding and decoding parallelizable in many ways
Goals

- Achieve a compression gain of 50% over H.264/AVC.
- x10 complexity max for encoder and x2/x3 max for decoder.
- Encoding and decoding parallelizable in many ways.
Profiles, levels, tiers

- A profile is a defined set of coding tools that can be used to create a bitstream that conforms to that profile.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Version 1</th>
<th>Version 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Main</td>
<td>Main 10</td>
</tr>
<tr>
<td>Bit depth</td>
<td>8</td>
<td>8 to 10</td>
</tr>
<tr>
<td>4:0:0 (Monochrome)</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>High precision weighted prediction</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Cross-component prediction</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Intra smoothing disabling</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Persistent Rice adaptation</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>RPDCM implicit/explicit</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Transform skip block sizes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>larger than 4x4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Transform skip context/rotation</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Extended precision processing</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>
Profiles, levels, tiers

- The HEVC standard defines two tiers, Main and High, and thirteen levels.
- A level is a set of constraints for a bitstream, e.g., max picture size, max sample rate.
- The tiers were made to deal with applications that differ in terms of their maximum bit rate.

<table>
<thead>
<tr>
<th>Level</th>
<th>Max Luma Picture Size (samples)</th>
<th>Max Luma Sample Rate (samples/s)</th>
<th>Main Tier Max Bit Rate (1000 bits/s)</th>
<th>High Tier Max Bit Rate (1000 bits/s)</th>
<th>Min Comp. Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>36864</td>
<td>552960</td>
<td>128</td>
<td>–</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>122880</td>
<td>3686400</td>
<td>1500</td>
<td>–</td>
<td>2</td>
</tr>
<tr>
<td>2.1</td>
<td>245760</td>
<td>7372800</td>
<td>3000</td>
<td>–</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>552960</td>
<td>16588000</td>
<td>6000</td>
<td>–</td>
<td>2</td>
</tr>
<tr>
<td>3.1</td>
<td>983040</td>
<td>33177600</td>
<td>10000</td>
<td>–</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>2228224</td>
<td>66846720</td>
<td>12000</td>
<td>30000</td>
<td>4</td>
</tr>
<tr>
<td>4.1</td>
<td>2228224</td>
<td>133693440</td>
<td>20000</td>
<td>50000</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>8912896</td>
<td>267386800</td>
<td>25000</td>
<td>100000</td>
<td>6</td>
</tr>
<tr>
<td>5.1</td>
<td>8912896</td>
<td>534773600</td>
<td>40000</td>
<td>160000</td>
<td>8</td>
</tr>
<tr>
<td>5.2</td>
<td>8912896</td>
<td>1069547520</td>
<td>60000</td>
<td>240000</td>
<td>8</td>
</tr>
<tr>
<td>6</td>
<td>35651584</td>
<td>1069547520</td>
<td>60000</td>
<td>240000</td>
<td>8</td>
</tr>
<tr>
<td>6.1</td>
<td>35651584</td>
<td>2139095040</td>
<td>120000</td>
<td>480000</td>
<td>8</td>
</tr>
<tr>
<td>6.2</td>
<td>35651584</td>
<td>4278190080</td>
<td>240000</td>
<td>800000</td>
<td>6</td>
</tr>
</tbody>
</table>
Performance

- Rate-distortion curves

![Rate-distortion curves for Johnny and KristenAndSara](image-url)
Performance

- HEVC calculated BD-BR savings (Compared to VP9 and x264 High Profile Encoders)

<table>
<thead>
<tr>
<th>Sequences/QPs</th>
<th>HEVC vs. VP9 (in %)</th>
<th>HEVC vs. x264 (in %)</th>
</tr>
</thead>
<tbody>
<tr>
<td>FourPeople</td>
<td>-31.6%</td>
<td>-27.6%</td>
</tr>
<tr>
<td>Johnny</td>
<td>-33.6%</td>
<td>-51.3%</td>
</tr>
<tr>
<td>KristenAndSara</td>
<td>-32.4%</td>
<td>-43.4%</td>
</tr>
<tr>
<td>Averages</td>
<td>-32.5%</td>
<td>-40.8%</td>
</tr>
</tbody>
</table>
Performance

- Bitrates reduction: same subjective visual quality

<table>
<thead>
<tr>
<th>Video coding standard</th>
<th>Average bit rate reduction compared to H.264/MPEG-4 AVC HP</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>480p</td>
</tr>
<tr>
<td>HEVC</td>
<td>52%</td>
</tr>
</tbody>
</table>
Performance

Bitrates reduction under same perceptual quality

1080p@25fps at 2.7 Mbits/s
Performance

- Bitrates reduction: same subjective visual quality
  - Profile: Main Still Picture

### Comparison of standards for still image compression based on equal PSNR and MOS

<table>
<thead>
<tr>
<th>Still image coding standard (test method)</th>
<th>Average bit rate reduction compared to</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>JPEG 2000</td>
</tr>
<tr>
<td></td>
<td>JPEG</td>
</tr>
<tr>
<td>HEVC (PSNR)</td>
<td>20.26%</td>
</tr>
<tr>
<td></td>
<td>61.63%</td>
</tr>
<tr>
<td>HEVC (MOS)</td>
<td>30.96%</td>
</tr>
<tr>
<td></td>
<td>43.10%</td>
</tr>
</tbody>
</table>
Performance

- Time complexity (vs. H.264)
  - Encoder: x4 ~ x6
  - Decoder: x1.5 ~ x2.5

(According to: Understanding the Significance of HEVC/H.265
https://www.youtube.com/watch?v=sB66Y_aE-58)

Decoding time distribution
Outline

• Introduction

• Technical Details
  • Coding structures
  • Intra prediction
  • Inter prediction
  • Transform
  • Scanning and Entropy Coding
  • In-loop filters
  • Parallel processing tools
Overview

- A block diagram of the HEVC framework
Coding Structures

- Coding Tree Unit (CTU)
  - Coding Tree Block (CTB)
  - 1 luma CTB + 2 chroma CTBs + associated syntax
- Coding Unit (CU)
  - Coding Block (CB)
- Prediction Unit (PU)
  - Prediction Block (PB)
- Transform Unit (TU)
  - Transform Block (TB)
Coding Structures

- Versus H.264 (1)
  - H.264: 16x16 macroblocks
    - Intra blocks: 16 4x4 sub-blocks or 1 16x16 block
    - Inter blocks: 16x16, 16x8, 8x16, 8x8 (and subpartitions)
  - HEVC: Coding tree blocks (CTBs)
    - 16x16, 32x32 or 64x64 blocks
    - Quadtree-like subpartitioning into coding blocks (CBs)
    - Minimum CB size: 8x8 (or larger if specified)
    - Chroma partitioned accordingly
Coding Structures

- Versus H.264 (1)
  - H.264: 16x16 macroblocks
    - Intra blocks: 16 4x4 sub-blocks or 1 16x16 block
    - Inter blocks: 16x16, 16x8, 8x16, 8x8 (and subpartitions)

- HEVC: Coding tree blocks (CTBs)
  - 16x16, 32x32 or 64x64 blocks
  - Quadtree-like subpartitioning into coding blocks (CBs)
  - Minimum CB size: 8x8 (or larger if specified)
  - Chroma partitioned accordingly
Coding Structures

• Versus H.264 (2)
  • H.264: Prediction and transform "static"
    • Prediction is coupled with block partition size
    • Transform size is always 4x4 (or adaptively 8x8 in High profile)
    • Intra/inter decision on (16x16) macroblock level
  • HEVC: Prediction and transform flexible
    • CBs split into (min. 4x4) prediction blocks (PBs)
    • CBs split into (min. 4x4) transform blocks (TBs)
    • TB structure may be further partitioned than the PB structure
    • Intra/inter decision on (min. 8x8) CB level
Coding Structures

BQMALL (832x480) frame 9 (P) – H.264 (left) vs. HEVC (right)
Outline

- Introduction
- Technical Details
  - Coding structures
  - Intra prediction
  - Inter prediction
  - Transform
  - Scanning and Entropy Coding
  - In-loop filters
  - Parallel processing tools
Intra prediction

- Intra Luma and Chroma prediction modes

  - Luma:
    - 35 modes: Planar + DC + 33 angular prediction modes for all block sizes
  
  - Chroma:
    - 5 modes: Derived mode (DM) + Planar + DC + Horizontal + Vertical
Intra prediction

- Intra prediction in HEVC
Intra prediction

- Versus H.264
  - H.264: Number of total modes depends on block size
    - 16 4x4 sub-blocks or 1 16x16 block
    - 4x4: DC or directional prediction (8 directions)
    - 16x16: DC, plane, horizontal or vertical prediction
    - 1 most probable model
  - HEVC: 35 modes in total
    - 32x32 down to 4x4 sub-blocks
    - DC, planar or directional prediction (33 directions)
    - 3 most probable modes
Outline

- Introduction
- Technical Details
  - Coding structures
  - Intra prediction
  - Inter prediction
  - Transform
  - Scanning and Entropy Coding
  - In-loop filters
  - Parallel processing tools
Inter prediction

- PB partitions for inter prediction
  - $2N \times 2N$, $N \times N$, $2N \times N$, $N \times 2N$, $2N \times nU$, $2N \times nD$, $nL \times 2N$, $nR \times 2N$
Inter prediction

- Versus H.264
  - H.264
    - 16x16, 16x8, 8x16 or 8x8 partitions
    - 8x4, 4x8 or 4x4 sub-partitions for 8x8 partitions
    - ¼ luma pixel accuracy
      - 6-tap filter for half pixel and averaging for quarter pixel
    - Use MV prediction to save bits
  - HEVC
    - More flexible, symmetric and asymmetric PB partitioning
    - ¼ luma pixel accuracy
      - 8-tap filter for half pixel and 7-tap filter for quarter pixel
    - Choose one MV from multiple candidates to save bits
Outline

• Introduction
• Technical Details
  • Coding structures
  • Intra prediction
  • Inter prediction
  • Transform
  • Scanning and Entropy Coding
  • In-loop filters
  • Parallel processing tools
Transform

- Core transforms: DCT based
  - 4x4, 8x8, 16x16, and 32x32
  - Near-orthogonal
  - Nested transforms
Transform

- Alternative 4x4 Discrete Sine Transform (DST)
  - 4x4 intra blocks, luma only
  - Reason: Residuals tend to increase with distance from boundary
  - 1% intra-only bit rate decrease (hardly any for larger sizes)

- Transform skipping mode
  - By-pass the transform stage
  - Most effective on “screen content”
  - 4x4 TBs only

- Scaling and quantization are similar to H.264
Transform

• Versus H.264
  • H.264: 4x4 Integer transform (DCT approximation)
    • Used for all block partitions and modes
    • Adaptive 8x8 integer transform in High Profile
  • HEVC: H.264-like transform for each transform block (TB)
    • More transform sizes: 4x4, 8x8, 16x16, 32x32
    • TBs must be squared
    • DST for 4x4 luma intra
Outline

• Introduction

• Technical Details
  • Coding structures
  • Intra prediction
  • Inter prediction
  • Transform
  • Scanning and Entropy Coding
  • In-loop filters
  • Parallel processing tools
Scanning and Entropy Coding

• Versus H.264
  • H.264
    • Scanning: zigzag scan
    • Entropy coding: CAVLC and CABAC
  • HEVC
    • Scanning: diagonal, horizontal, vertical
    • Entropy coding: CABAC only
      • Similar to CABAC in H.264
      • Reduced data dependency, easier to parallelize
Outline

- Introduction
- Technical Details
  - Coding structures
  - Intra prediction
  - Inter prediction
  - Transform
  - Scanning and Entropy Coding
  - In-loop filters
  - Parallel processing tools
In-loop filters

- Two processing steps:
  - Deblocking filtering
  - Sample adaptive offset (SAO)
In-loop filters

- Sample Adaptive Offset (SAO)
  - A nonlinear filtering operation which modifies the samples after the deblocking filtering to enhance quality in both smooth areas and around edges
  - Adds (conditionally) an offset value to each sample based on values in look-up tables transmitted by the encoder
Outline

• Introduction

• Technical Details
  • Coding structures
  • Intra prediction
  • Inter prediction
  • Transform
  • Scanning and Entropy Coding
  • In-loop filters
  • Parallel processing tools
Parallel processing tools

- Wavefront parallel processing (WPP)
  - Decoding of each CTU row can be parallelized

![Diagram showing multi-thread decoding with wavefronts](image)

Thread 1
Thread 2
Thread 3
Thread 4
Parallel processing tools

- **Tiles**
  - Rectangular region of CTUs
  - Parallelization for both encoder and decoder

![Diagram of tiles division](image)
References

- [5] InterDigital, “HEVC Overview”, online PPT.
References

- [8] Nam Ling, et al., “Next Generation Video Coding - HEVC with a Special Look at Intra Prediction”, online PPT.