ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

Multimedia Technology

Section # 15: Video coding: H.264

Instructor: George Xylomenos

Department: Informatics

Contents

- Introduction
- Network support
- Video coding: prediction
- Video coding: transform, quantization, entropy
- Profiles, levels and recovery
- Scalable coding

ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

Introduction

Class: Multimedia Technology, **Section # 15:** Video coding: H.264

Instructor: George Xylomenos, **Department:** Informatics

What is H.264 (1 of 3)

- Common standard from two organizations
  - ITU-T H.264/AVC (2004)
  - ISO/IEC MPEG-4 Part 10
    - Very different from MPEG-4 Part 2!
- More efficient compressions
  - Twice as much as MPEG-2
    - But with triple the computation load
    - Despite a simplified DCT!

What is H.264 (2 of 3)

- More friendly to different networks
  - Applicable to conferencing (like H.263)
  - Applicable to distribution (like MPEG-4 Part 2)
  - More resilient to errors
- More flexible
  - Random access and chapters
  - Fast/slow motion forwards/backwards

What is H.264 (3 of 3)

- Evolution of previous standards
  - Concepts from both MPEG-4 Part 2 and H.263
  - Unifies the previous MPEG and H.26x standards
  - Next: H.265/HEVC (2013), H.266/VVC (2020)
- Applications
  - Bluray disks
  - HDTV (e.g. in DVB-T)
  - Web Video

Network support

Class: Multimedia Technology, **Section # 15:** Video coding: H.264

Instructor: George Xylomenos, **Department:** Informatics

Network support (1 of 5)

- Standard split in two layers
  - Video Coding Layer (VCL)
  - Network Abstraction Layer (NAL)
- NAL: adaptation to various transports
  - RTP/UDP/IP for IP networks
  - MPEG-2 TS for HDTV
  - Files for storage and exchange
  - H.32x standards for conferencing

Network support (2 of 5)

- Bit stream broken into NAL Units (NALU)
  - Each starts with a 1 byte header
  - Adaptation for packet network
    - Direct encapsulation in packets (e.g., RTP)
    - Fragmentation or packing of NALUs
  - Adaptation for byte stream networks
    - Each NALU starts with 3 framing bytes
    - Byte stuffing to avoid erroneous frames

Network support (3 of 5)

- Two NALU types
  - VCL: contain video info
  - Non-VCL: parameter sets, timing info
- Parameter sets
  - Common parameters for many VCL NALUs
  - Sent before image data
  - Can be transmitted with enhanced resilience
    - Either using a separate reliable channel
    - Or with additional protection (e.g., repetition coding)

Network support (4 of 5)

- Parameter sets
  - Two levels: sequence or frame
  - Frame parameters: reference sequence parameters
  - VCL NALU: reference frame parameters
- Access unit
  - All NALUs related to a video frame
    - Timing and location info
    - Coded image data
    - Image data encoded at lower quality

Network support (5 of 5)

- Video sequence
  - A sequence of access units
  - Using the same parameter set
  - Starting with frame that does not reference others
    - Can be independently decoded
  - Similar to GOPs in MPEG-2

ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

Video coding: prediction

Class: Multimedia Technology, **Section # 15:** Video coding: H.264

Instructor: George Xylomenos, **Department:** Informatics

Slices (1 of 4)

- Video Coding Layer (VCL)
  - Based on known coding techniques
    - Extensions of H.263 and MPEG-4 Part 2
  - Plus, many small improvements
    - That all contribute to improved coding
- Breaking down a frame into MBs
  - The MBs are organized into slices
  - The number of MBs per slice is arbitrary
    - Not one slice per row of MBs as in MPEG-2

Slices (2 of 4)

- What are slices used for?
  - Slices are autonomously decoded
  - They allow sync after errors
    - They start with a known bit sequence
  - They adapt to the packet size
    - Ideally: one slice per packet
    - Packet loss→slice loss
  - More important slices can be repeated

Slices (3 of 4)

- Flexible Macroblock Ordering (FMO)
  - Slices where MBs are NOT consecutive
    - Map used to assign MBs to slice
  - Different types of ordering
    - MB interleaving to increase reliability
    - Separation of important MBs for extra resilience

Slices (4 of 4)

- Adaptive Slice Ordering (ASO)
  - Transmission of slices in any order
- Slice types
  - I Slice: only non predicted MBs
  - P Slice: adds MBs with single prediction
  - B Slice: adds MBs with double prediction
  - SP/SI Slices: used for random access

Intraframe coding (1 of 3)

- Intraframe coding: I-macroblocks
- Allow prediction from areas <u>in the same frame</u>
  - As long as they have already been encoded
    - But: these areas may be interframe coded!
  - No prediction outside the slice allowed
- Limited intraframe coding
  - Prediction only from intraframe coded areas
  - An I-frame can only have this type of MBs

Intraframe coding (2 of 3)

- Prediction in 4x4 or 16x16 areas
- 4x4 for higher detail
  - Multiple prediction types
    - Always within the same slice!
  - Average value of pixels above / left
  - Corresponding pixels above the block
  - Corresponding pixels left of the block
  - Generalized with diagonals

Intraframe coding (3 of 3)

- 16x16 for background or simpler areas
  - Fewer prediction types
  - Average, pixels above / left
- Ends with a DCT transform
  - Of the differences from the prediction
- Uncompressed MBs (PCM)
  - Pixel values without any compression
  - For very complex blocks or very high detail

Interframe coding (1 of 6)

- Interframe coding: P-macroblocks
- Different block sizes possible
  - Sized 16x16, 16x8, 8x16 or 8x8
    - Variations of the 16x16 MB
  - 8x8 blocks can be divided into 8x4, 4x8 or 4x4
    - Further divisions of the 8x8 block
  - Each block has its own motion vector

Interframe coding (2 of 6)

- Various prediction types
  - Regular GOP with IPB frames
  - No B-frames for conferencing
  - Multiple reference frames
    - The motion vectors include the reference
  - Hierarchical prediction
    - Gradual decoding and prediction
    - Allows prediction from B-blocks to B-blocks

Interframe coding (3 of 6)

- Non integer motion vectors
  - 1, ½ or ¼ pixel resolution
- References with ½ pixel accuracy
  - The reference is formed via a filter
  - Linear combination of +/-3 neighboring pixels
- References with ¼ pixel accuracy
  - Average of neighboring pixels
    - Pixels in integer and ½ pixel distance

Interframe coding (4 of 6)

- Motion vectors outside the frame
  - Assumes that the edge values are repeated
- Differential coding of motion vectors
  - Prediction only within a slice
  - May be directional or average
- Simple prediction (P_Skip)
  - Nearly nothing transmitted!
  - Same pixel values as reference
  - Same motion vector as previous block

Interframe coding (5 of 6)

- Interframe coding: B-macroblocks
  - Generalization of B-frames
    - Can be used as references for prediction
    - Based on a combination of predictors
  - Four prediction types
    - Two frame lists, combination and direct
    - Direct is based on simple prediction
  - Allows different block sizes

Interframe coding (6 of 6)

- Combination of predictors
  - Motion vectors also have weights
    - We find the reference area
    - We combine the areas based on the weights

ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

Video coding: transform, quantization, entropy

Class: Multimedia Technology, **Section # 15:** Video coding: H.264

Instructor: George Xylomenos, **Department:** Informatics

Transform (1 of 2)

- Integer DCT used
  - Matrix multiplication with integer values
    - Large speedup of calculations
    - No rounding errors
    - Reversible without any loss
  - Based on 16 bit arithmetic
    - Previous methods used 32 bit arithmetic
    - Less space and smaller computational cost

Transform (2 of 2)

- Optionally in 4x4 instead of 8x8 blocks
  - Allows more ways to predict a block
  - Which leads to smaller differences
  - Improves quality at the edges of blocks
  - Simpler calculations with smaller numbers
- DC coefficients are transformed again
  - New matrix with the DCs of each 4x4 block
  - Improves compression in simple areas

Quantization

- Based on a single scaling value
  - 52 different values (0-51)
  - Each value is a quantization step
  - The matrix is generated from the scaling value
- Logarithmic rather than linear quantization
  - Every 6 steps the quantizer is doubled
  - Each step increases the quantizer by 12%

Entropy coding (1 of 3)

- Option 1: variable length coding
- ExpGolomb code for simple data
  - Can encode any number
  - Fewer bits for smaller numbers
  - Format: prefix1suffix
  - The prefix (zeroes) encodes the length
  - The suffix encodes the value

Entropy coding (2 of 3)

- Adaptive (CAVLC) code for the coefficients
  - Multiple tables for each type
  - Depending on the value of the block
    - How many non-zero values
    - How many ones at the end
    - What other numbers exist
    - How many zeroes between them
  - Better compression compared to single table

Entropy coding (3 of 3)

- Option 2: arithmetic coding
  - Also adaptive (CABAC)
  - We first turn everything to an integer
  - Then we choose a prediction model
    - Essentially, a model for the probabilities
  - Encoding based only on integer arithmetic
    - Uses scaling to avoid floating point
  - Improves compression by 5-15%

Deblocking filter

- At low bit rates, artefacts appear
  - The edges of the block tend to the average value
  - Visible changes at block ends
- The filter reduces these artefacts
  - Looks at pixels in neighboring blocks
  - Differences > quantizer are not changed
  - Smaller differences are filtered (smoothed)
  - Allows 5-10% more encoding for the same quality

Profiles, levels and recovery

Class: Multimedia Technology, **Section # 15:** Video coding: H.264

Instructor: George Xylomenos, **Department:** Informatics

Capability sets (1 of 2)

- Capability sets of H.264/AVC
  - 0: I and P blocks, Huffman coding
  - 1: non continuous slices, slice repetition
  - 2: SP/SI blocks, data partitioning
  - 3: B blocks , weighted prediction
  - 4: arithmetic coding

Capability sets (2 of 2)

- Profile: which capability sets are allowed
  - Baseline: sets 0 and 1 (reliability)
  - Main: sets 0, 3 and 4 (performance)
  - Extended: sets 0, 1, 2 and 3
- Level: values of basic parameters
  - 15 levels, applicable to all profiles
  - Resolution, MB rate, bit rate

Error recovery (1 of 4)

- The encoder provides many tools
- Two types of intracoded MBs
  - Restricted MBs: prediction only from I-MBs
- Packet-sized slices
  - Limits the damage from packet loss
- Selection of the reference frame
  - Allows bypassing lost blocks

Error recovery (2 of 4)

- Parameter sets
  - Can be sent with additional error correction
- Flexible MB ordering
  - Interleaving spreads the effect of errors
- Slice repetition
  - Important parts of the frame can be repeated
  - They can even be re-sent at lower quality

Error recovery (3 of 4)

- Data are split in three sets
  - A: MB types used, motion vectors, quantizer
  - B: intraframe-coded MBs
  - C: interframe-code MBs
  - Descending order of importance
    - Set A is required for B and C
    - Set A allows error concealment
    - Can be handled differently by the network

Error recovery (4 of 4)

- NALU marking
  - Each NALU has a priority in the header
    - The network can select what to drop
    - Works very well with data splitting!
  - Forbidden bit: erroneous data
    - For example, packets with bit errors
    - Allows the decoder to act accordingly
    - Recover whatever possible, conceal errors

ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

Scalable coding

Class: Multimedia Technology, **Section # 15:** Video coding: H.264

Instructor: George Xylomenos, **Department:** Informatics

Scalability (1 of 5)

- Scalable Video Coding (SVC) extensions
  - Three scalable levels
    - Multiple resolutions/qualities/frame rates
  - Full or partial transmission and decoding
  - Easy extraction of non-scalable bit stream
  - Small bitrate overhead for scalability
  - Small processing overhead for scalability
    - Compared to MPEG-2 and MPEG-4 part 2

Scalability (2 of 5)

- SVC goals
  - Support heterogeneous terminals
  - Support heterogeneous networks
- Scalability options
  - Temporal (frame rate)
    - Supported by baseline H.264/AVC
  - Spatial (resolutions)
    - Multiple resolutions in the same flow
  - Quality (image precision)

Scalability (3 of 5)

- Layered coding
  - The bitstream contains numbered layers
  - Layer 0 does not need special decoding
    - Plain H.264/AVC decoder is enough
  - Layers >0 improve resolution or quality
    - Compared to the previous layers
    - Each frame can have different numbers of layers

Scalability (4 of 5)

- Prediction across layers
  - MB or sub-MB based
  - Simple case: same as in reference layer
    - Intraframe: same as in reference layer
    - Interframe: same prediction as in reference layer
    - Reference areas and motion vectors are scaled
  - Otherwise, encode difference from reference
    - Reference area scaled before calculating difference

Scalability (5 of 5)

- Common decoding for all layers
  - Not separate per layer
- Temporal scaling
  - Increased frame rate in each layer
  - Careful selection of reference frames
    - Limited to lower layers

End of Section # 15

Class: Multimedia Technology, **Section # 15:** Video coding: H.264

Instructor: George Xylomenos, **Department:** Informatics