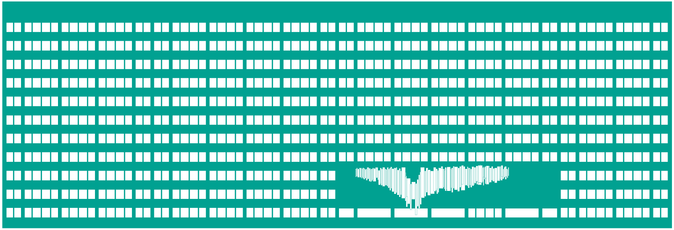
VŠB TECHNICKÁ UNIVERZITA OSTRAVA

VSB TECHNICAL UNIVERSITY OF OSTRAVA



www.vsb.cz

Video Compression

H.264

Michal Vasinek

VSB – Technical University of Ostrava

name.surname@vsb.cz

May 23, 2019

VSB TECHNICAL UNIVERSITY OF OSTRAVA

Overview

- Motivation
- Approach
- Basic video frames
- H.264

Internet traffix

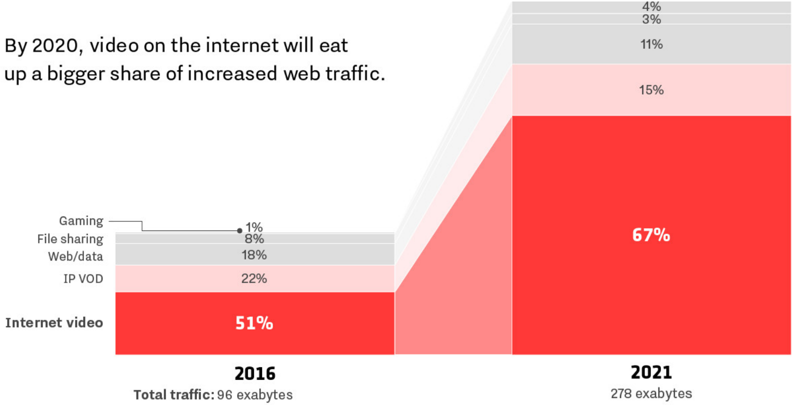


Figure: Internet traffic prediction - source Cisco

Motivation

Consider the example:

- 30 second video: full HD resolution 1920x1080 at 30 fps.
- Decoding into RGB - 24 bits per pixel -> 6.2 MB per frame.
- Result 6.2 MB x 30 sec x 30 fps = 5.2GB
- Modern smartphones 128 GB internal memory => 750 seconds of video or 12.5 minutes.
- Idea 1: compress each frame using JPEG => 531 MB, 128 minutes or 2 hours.
- H.264 - video file size: 65.4 MB => approximately 16 hours of video.

H264/AVC video compression

- AVC - advanced video coding
- Also called MPEG4 Part 10
- Applications:
  - Blue Ray
  - HD streaming on internet(Youtube, Netflix, Vimeo, iTunes)
  - HD video in smartphones
  - Public TV broadcast in Europe.
- Benefit: higher compression ratios than MPEG2 or MPEG4
- Costs: higher decoding complexity.
- Main idea: **trades off more computing for requiring less bandwidth/storage**

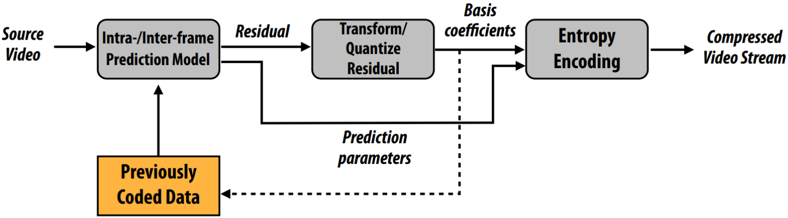
Hardware support

- Support for H.264 encode/decode on most modern processors.
- Hardware encoding/decoding support existed in modern Intel CPUs since Sandy Bridge architecture(2009).
- Modern operating systems expose hardware encode/decode support through hardware-accelerated APIs: DirectShow/DirectX (Windows), AVFoundation(iOS)

Video compression: main ideas

- Compression is about exploiting redundancy in signal:
  - Intra-frame redundancy - value of pixels in neighboring regions of a frame are good predictor of values for other pixels in the frame (**spatial redundancy**)
  - Inter-frame redundancy - pixels from nearby frames in time are a good predictor for the current frame's pixels (**temporal redundancy**)

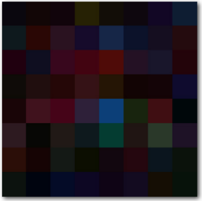
H.264 - overview



Residual: difference between predicted pixel values and input video pixel values

Residual: difference between compressed image and original image

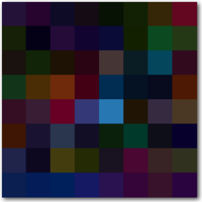
Compressed pixels
(JPEG quality level 6)



Residual
(amplified for visualization)

Original pixels

Compressed pixels
(JPEG quality level 2)



Residual
(amplified for visualization)

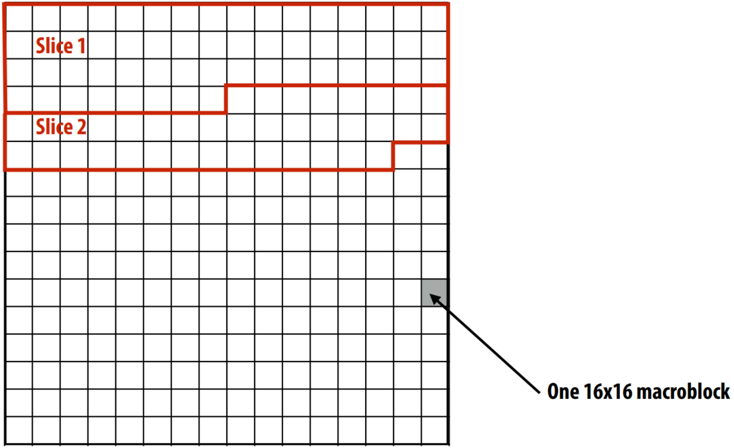
H.264 - what is defined

- H.264 standard defines how to represent and decode video
- H.264 does not define how to encode video (this is left up to implementations)
- H.264 has many profles

H.264 - frame

- Frame consists of 16x16 pixels macroblocks.
- 4:2:0 chroma subsampling:
  - Lumina component (brightness) in full resolution => 16x16 pixels.
  - Chroma components at half resolution => 8x8 pixels.
- Macroblocks organized into slices - slice is a sequence of macroblock in frame scan order.
- Slices can be decoded independently.

H.264 - slices



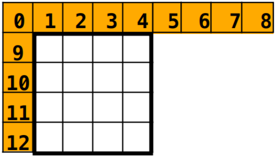
Macroblock reconstruction:

- Prediction is based on already decoded samples in macroblocks from the same frame (intra-frame prediction) or from other frames (inter-frame prediction) .
- Correcting the prediction with a residual stored in the video stream.

H.264 - prediction

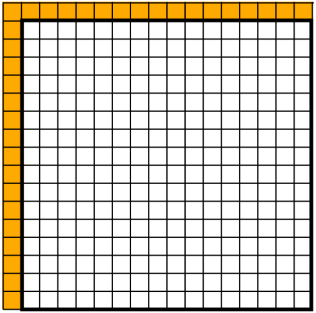
- **I-macroblock** - macroblocks are predicted from samples in previous macroblocks in the same slice of the current frame.
- **P-macroblock** - macroblocks are predicted from samples from one other frame.
- **B-macroblock** - macroblocs are predicted by a weighted combination of multiple predictions from samples from other frames (past and future).

- **Modes for predicting the 16x16 luma (Y) values: ***
  - **Intra_4x4 mode: predict 4x4 block of samples from adjacent row/col of pixels**
  - **Intra_16x16 mode: predict entire 16x16 block of pixels from adjacent row/col**
  - **I_PCM: actual sample values provided**



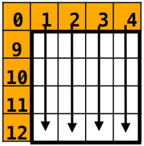
Intra_4X4

Yellow pixels: already reconstructed (values known)
White pixels: 4x4 block to be reconstructed

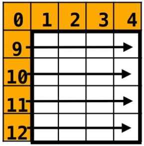


Intra_16x16

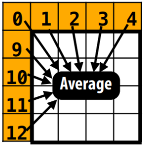
- **Nine prediction modes (6 shown below)**
  - **Other modes: horiz-down, vertical-left, horiz-up**



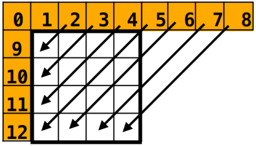
Mode 0: vertical (4x4 block is copy of above row of pixels)



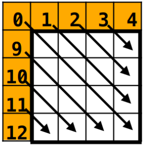
Mode 1: horizontal (4x4 block is copy of left col of pixels)



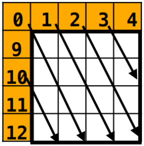
Mode 2: DC (4x4 block is average of above row and left col of pixels)



Mode 3: diagonal down-left (45°)

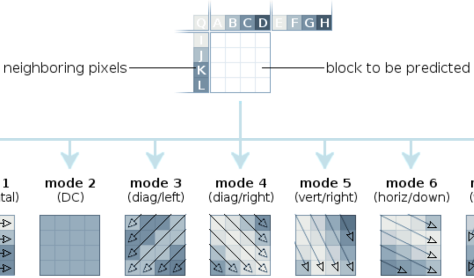


Mode 4: diagonal down-right (45°)



Mode 5: vertical-right (26.6°)

H.264 - I-macroblock, 4x4 intra mode

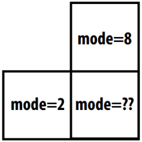


AVC/H.264 intra prediction modes

H.264 - I-macroblock, 4x4 intra mode

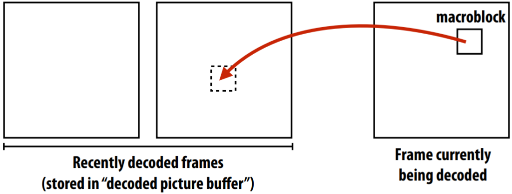
- **Intra-prediction of chroma (8x8 block) is performed using four modes similar to those of intra_16x16 (except reordered as: DC, vertical, horizontal, plane)**

- **Intra-prediction scheme for each 4x4 block within macroblock encoded as follows:**
  - **One bit per 4x4 block:**
    - **if 1, use <u>most probable</u> mode**
      - **Most probable = lower of modes used for 4x4 block to left or above current block**
    - **if 0, use additional 3-bit value `rem_intra4x4_pred_mode` to encode one of nine modes**
      - **if `rem_intra4x4_pred_mode` is smaller than most probable mode, use mode given by `rem_intra4x4_pred_mode`**
      - **else, mode is `rem_intra4x4_pred_mode+1`**



Inter-frame prediction (P-macroblock)

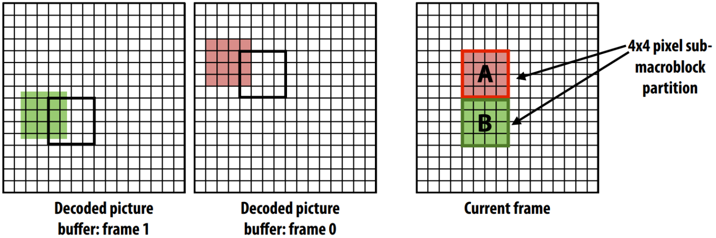
- **Predict sample values using values from a block of a <u>previously decoded frame</u> ***
- **Basic idea: current frame formed by translation of pixels from temporally nearby frames (e.g., object moved slightly on screen between frames)**
  - **"Motion compensation": use of spatial displacement to make prediction about pixel values**



*** Note: "previously decoded" does not imply source frame must come before current frame in the video sequence. (H.264 supports decoding out of order.)**

P-macroblock prediction

- Prediction can be performed at macroblock or sub-macroblock granularity
  - Macroblock can be divided into 16x16, 8x16, 16x8, 8x8 "partitions"
  - 8x8 partitions can be further subdivided into 4x8, 8x4, 4x4 sub-macroblock partitions
- Each partition predicted by sample values defined by: (reference frame id, motion vector)



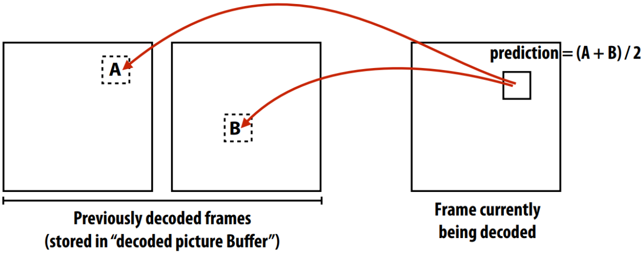
Block A: predicted from (frame 0, motion-vector = [-3, -1])

Block B: predicted from (frame 1, motion-vector = [-2.5, -0.5]) ← Note: non-integer motion vector

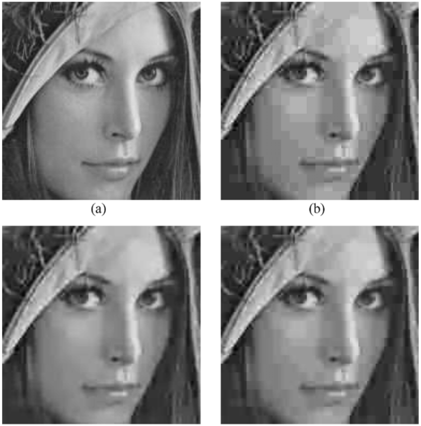
H.264 - P-macroblock motion vector example

Inter-frame prediction (B-macroblock)

- **Each partition predicted by up to two source blocks**
  - **Prediction is the average of the two reference blocks**
  - **Each B-macroblock partition stores two frame references and two motion vectors (recall P-macroblock partitions only stored one)**



- **Deblocking**
  - Blocking artifacts may result as a result of macroblock granularity encoding
  - After macroblock decoding is complete, optionally perform smoothing filter across block edges.



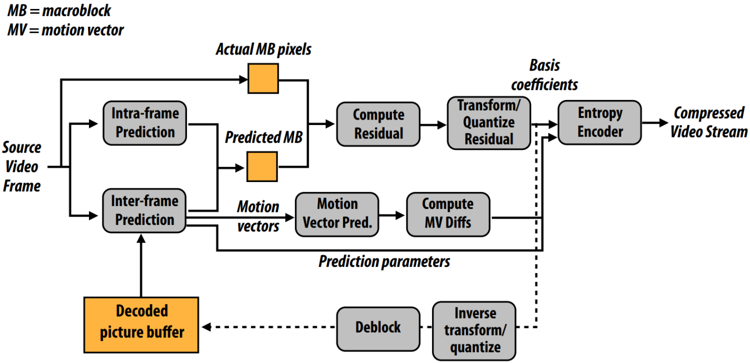
(a) (b) (c) (d)

H.264 - General encoding inter-predicted macroblock

- **Inputs:**
  - **Current state of decoded picture buffer (state of the decoder)**
  - **16x16 block of input video to encode**
- **General steps: (need not be performed in this order)**
  - **Resample images in decoded picture buffer to obtain 1/2, and 1/4, 1/8 pixel resampling**
  - **Choose prediction type (P-type or B-type)**
  - **Choose reference pictures for prediction**
  - **Choose motion vectors for each partition (or sub-partition) of macroblock**
  - **Predict motion vectors and compute motion vector difference**
  - **Encode choice of prediction type, reference pictures, and motion vector differences**
  - **Encode residual for macroblock prediction**
  - **Store reconstructed macroblock (post deblocking) in decoded picture buffer to use as reference picture for future macroblocks**

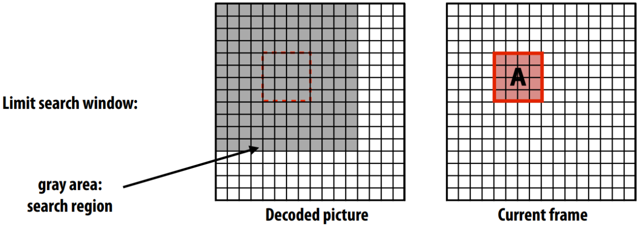
Coupled decisions

H.264 - Encoding summary



H.264 - Encoding motion estimation

- Encoder must find reference block that predicts current frame's pixels well.
  - Can search over multiple pictures in decoded picture buffer + motion vectors can be non-integer (huge search space)
  - Must also choose block size (macroblock partition size)
  - And whether to predict using combination of two blocks
  - Literature is full of heuristics to accelerate this process
    - Remember, must execute motion estimation in real-time for HD video (1920x1080), on a low-power smartphone



H.265 - HEVC

High efficiency video coding (HEVC).

- **Standard ratified in 2013**
- **Goal: ~2X better compression than H.264**
- **Main ideas:**
  - **Macroblock sizes up to 64x64**
  - **Prediction block size and residual block sizes can be different**
  - **35 intra-frame prediction modes (recall H.264 had 9)**
  - **...**

Thank you for your attention

Michal Vasinek

VSB – Technical University of Ostrava

name.surname@vsb.cz

May 23, 2019

VSB TECHNICAL UNIVERSITY OF OSTRAVA