Compression of Video Data

Computer Science

Philip Geißler

May 29, 2017

Carl-Zeiss-Gymnasium Jena

Table of contents

1. Introduction

2. Ideas

3. Meet: H.264, MPEG 1-4, VP9

4. Techniques of advanced Videocompression

5. Sources

Introduction

Practical Example

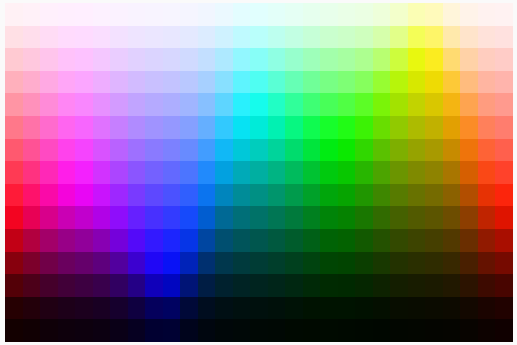
Funny Video #3429

Key Data	Factor
Full HD 1920×1080	2.073.600
13 Minutes long	780
16 Bit Color Depth	16
30 fps	30
Uncompressed	≈ 97.000 MB
Compressed	≈ 247 MB

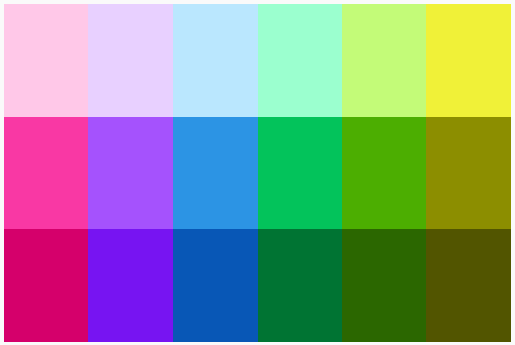
Ideas

Size Reduction

580×300 p



29×15 p



6×3 p

Table 1: Pictures with differently strong reduced sizes.

Examples:

- 480p, 360p, 240p, 144p Youtube videos
- low quality video previews

Problems:

- loss of detail
- blocky appearance

Frame-frequency Reduction

Table 2: Moving circles with different updating frequencies

Examples:

- 30fps Youtube videos
- 24fps Blockbusters

Problems:

- content-stuttering
- jumps in position for fast moving Objects

Reduction of Information per Frame - Interlacing

Table 3: interlaced Frame and deinterlaced counterpart

Interlaced broadcasting updates every second line/column of pixels

Examples:

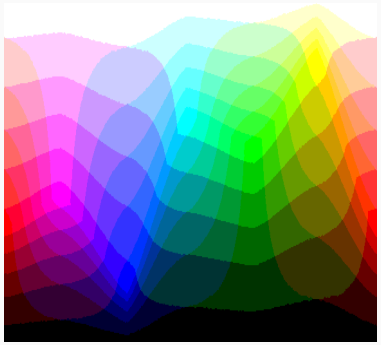
- 1080i TV
- image/gif previews

Problems:

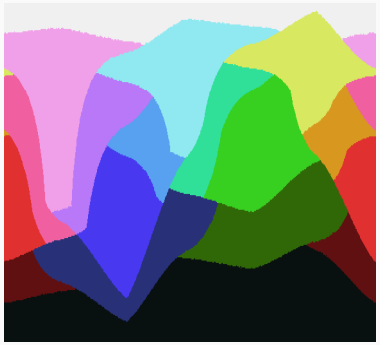
- edgecombing for fast moving objects
- artifacs in deinterlaced videos

Color-depth Reduction

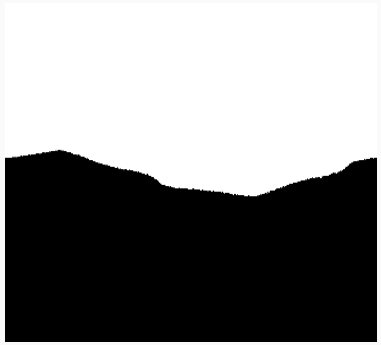
16 bit



8 bit



4 bit



1 bit

Table 4: Pictures with differently strong reduced color-depths

Examples:

- 8-Bit graphics
- Youtube videos of all qualities

Problems:

- sharp edges instead of smooth gradients
- less acute representation of dark areas

Redundancy Reduction

Table 5: Picturesequence with almost no additional information

Advantages:

- huge loss of data with little loss of information

Goals:

- reutilization of nonchanging data
- compression of large areas with low information
- focus on detailed high-contrast areas

Meet:
H.264, MPEG 1-4, VP9

Video & Audio compression Standards

H.264

- developed by ”Microsoft”, ”Cisco” and the ”Fraunhofer Institute of telecommunications”
- use of AVC1, DAVC, x264 and VSSH encoders

Mepg 1-4

- developed by the ”Moving Picture Experts Group”
- 1993 MPEG1 for video-CDs
- 1995 MPEG2 for DVDs and TV
- 2002 MPEG4 for internet video- and audio data

VP9

- developed by ”Google Inc.”
- mainly used for Youtube videos

Techniques of advanced Videocompression

Video Compression Techniques

development of two subtypes
cooperation for maximum compression

Type:	
Task:	
Example:	

Video Compression Techniques

development of two subtypes
cooperation for maximum compression

Type:	Intraframe Spatial Compression
Task:	single frame compression
Example:	JPEG

Video Compression Techniques

development of two subtypes
cooperation for maximum compression

Type:	Intraframe Spatial Compression	Interframe Temporal Compression
Task:	single frame compression	video compression through keyframes
Example:	JPEG	MPEG

JPEG Compression

- Chroma Subsampling

JPEG Compression

- Chroma Subsampling
- Macroblocks

JPEG Compression

- Chroma Subsampling
- Macroblocks
- Discrete Cosine Transformation

JPEG Compression

- Chroma Subsampling
- Macroblocks
- Discrete Cosine Transformation
- Quantization

JPEG Compression

- Chroma Subsampling
- Macroblocks
- Discrete Cosine Transformation
- Quantization
- Entropy Encoding

JPEG Compression - Chroma Subsampling

RGB to 3 new channels

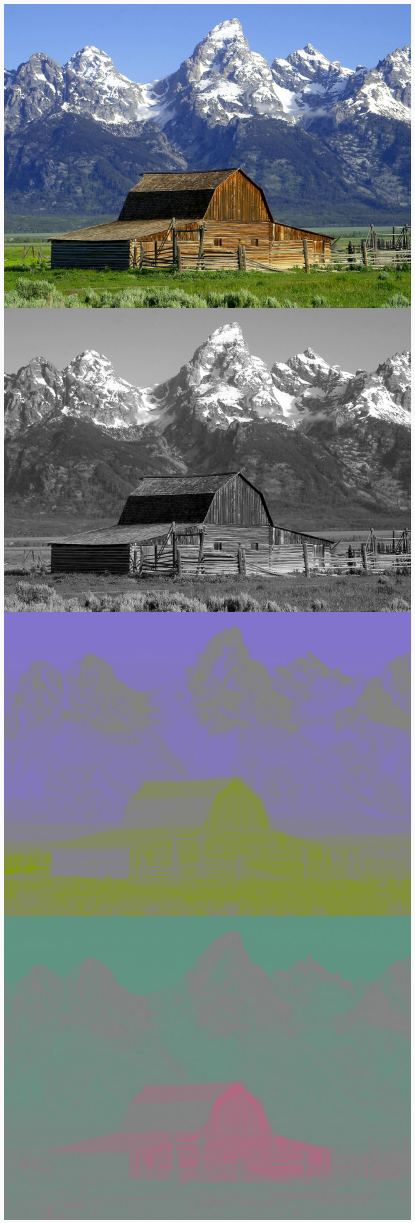
- Y’ - Luminance
- Cb - Chrominance 1
- Cr - Chrominance 2

biological facts

- humans are more sensitive to brightness than to color
- ∴ Chrominance can be subsampled

normal sampling rates

- 4:4:4 - no compression
- 4:2:2 - TV
- 4:2:0 - JPEG and MPEG



JPEG Compression - Macroblocks

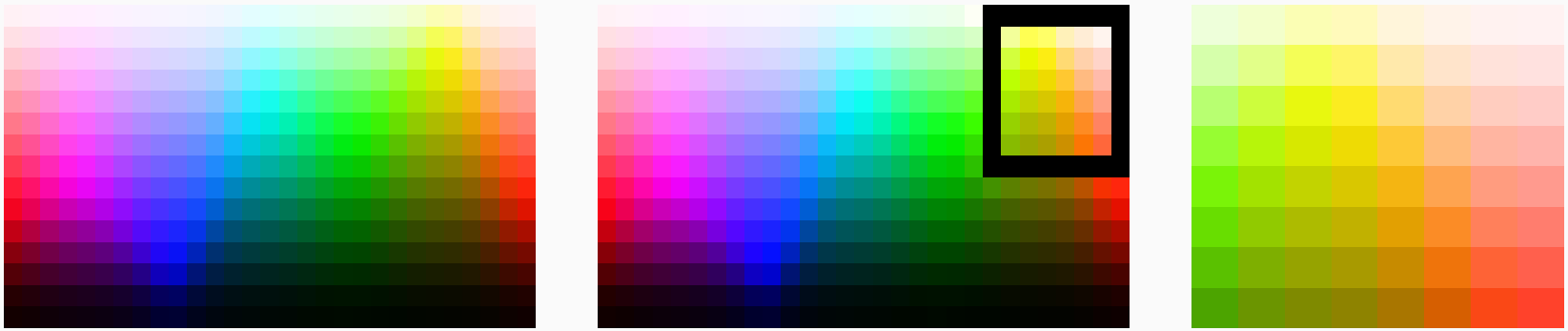
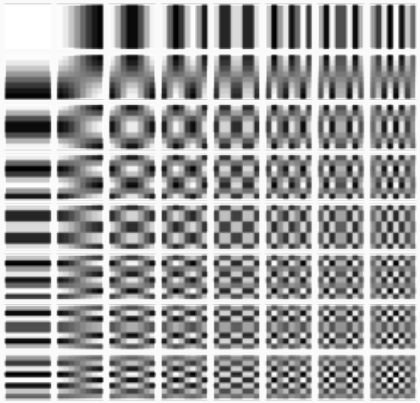


Table 6: extraction of a macroblock from an Image

Pictures get sectioned into 8×8 macroblocks
macroblocks will get compressed independently of one another

JPEG Compression - Discrete Cosine Transformation

248	248	247	246	244	244	244	244
239	239	238	235	232	231	231	231
228	227	226	222	218	216	214	214
215	213	209	205	200	197	194	195
199	194	190	186	181	176	174	174
179	173	169	166	161	155	151	151
155	150	147	144	140	133	127	127
132	128	124	122	119	112	105	103



$$G = \begin{bmatrix} -415.38 & -30.19 & -61.20 & 27.24 & 56.12 & -20.10 & -2.39 & 0.46 \\ 4.47 & -21.86 & -60.76 & 10.25 & 13.15 & -7.09 & -8.54 & 4.88 \\ -46.83 & 7.37 & 77.13 & -24.56 & -28.91 & 9.93 & 5.42 & -5.65 \\ -48.53 & 12.07 & 34.10 & -14.76 & -10.24 & 6.30 & 1.83 & 1.95 \\ 12.12 & -6.55 & -13.20 & -3.95 & -1.87 & 1.75 & -2.79 & 3.14 \\ -7.73 & 2.91 & 2.38 & -5.94 & -2.38 & 0.94 & 4.30 & 1.85 \\ -1.03 & 0.18 & 0.42 & -2.42 & -0.88 & -3.02 & 4.12 & -0.66 \\ -0.17 & 0.14 & -1.07 & -4.19 & -1.17 & -0.10 & 0.50 & 1.68 \end{bmatrix}$$

Table 7: discrete cosine transformation of a macroblock

macroblock matricies get normalized (-128)
DCT function gets applied

JPEG Compression - Quantization

$$G = \begin{bmatrix} -415.38 & -30.19 & -61.20 & 27.24 & 56.12 & -20.10 & -2.39 & 0.46 \\ 4.47 & -21.86 & -60.76 & 10.25 & 13.15 & -7.09 & -8.54 & 4.88 \\ -46.83 & 7.37 & 77.13 & -24.56 & -28.91 & 9.93 & 5.42 & -5.65 \\ -48.53 & 12.07 & 34.10 & -14.76 & -10.24 & 6.30 & 1.83 & 1.95 \\ 12.12 & -6.55 & -13.20 & -3.95 & -1.87 & 1.75 & -2.79 & 3.14 \\ -7.73 & 2.91 & 2.38 & -5.94 & -2.38 & 0.94 & 4.30 & 1.85 \\ -1.03 & 0.18 & 0.42 & -2.42 & -0.88 & -3.02 & 4.12 & -0.66 \\ -0.17 & 0.14 & -1.07 & -4.19 & -1.17 & -0.10 & 0.50 & 1.68 \end{bmatrix}$$

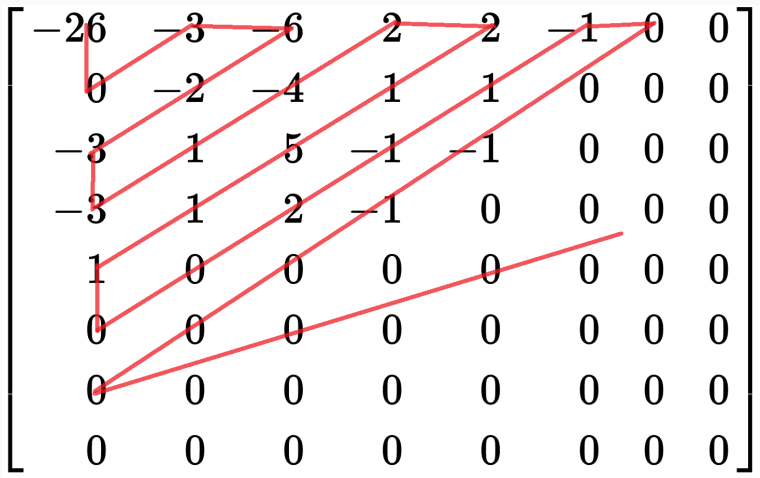
$$\begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix}$$

$$\begin{bmatrix} -26 & -3 & -6 & 2 & 2 & -1 & 0 & 0 \\ 0 & -2 & -4 & 1 & 1 & 0 & 0 & 0 \\ -3 & 1 & 5 & -1 & -1 & 0 & 0 & 0 \\ -3 & 1 & 2 & -1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Table 8: Quantization of a macroblock

$$M_{out}(x, y) = \left\lfloor \frac{M_{in}(x, y)}{M_{quantization}(x, y)} \right\rfloor (+128)$$

JPEG Compression - Entropy Encoding



$$-26, 0, -3, -6, \cdots -1, [0, 0, \cdots]$$

$$2, 125, 128, 125, \cdots 127, 40 \times 0$$

$$10100010001111110100110\cdots$$

tailcompression through run-length encoding
another compression layer through huffman encoding

MPEG Compression

3 categories of frames

- I - Frame

MPEG Compression

3 categories of frames

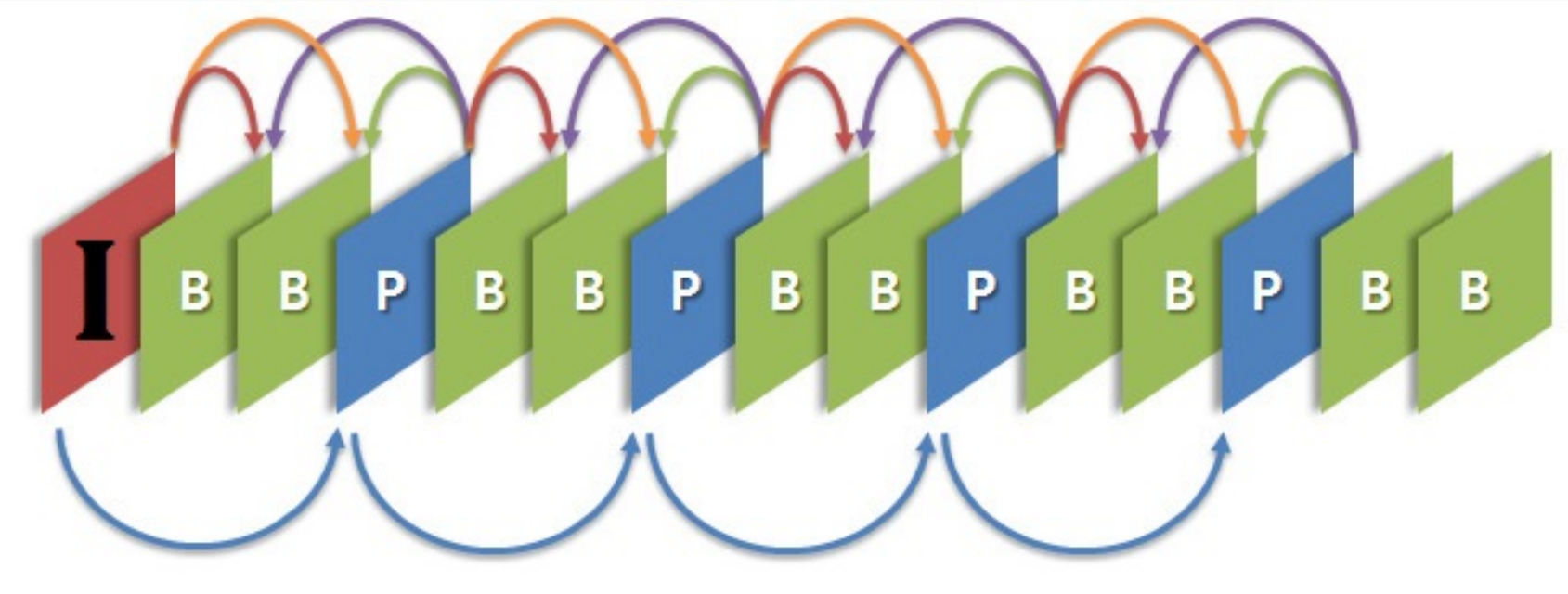
- I - Frame
- P - Frame

MPEG Compression

3 categories of frames

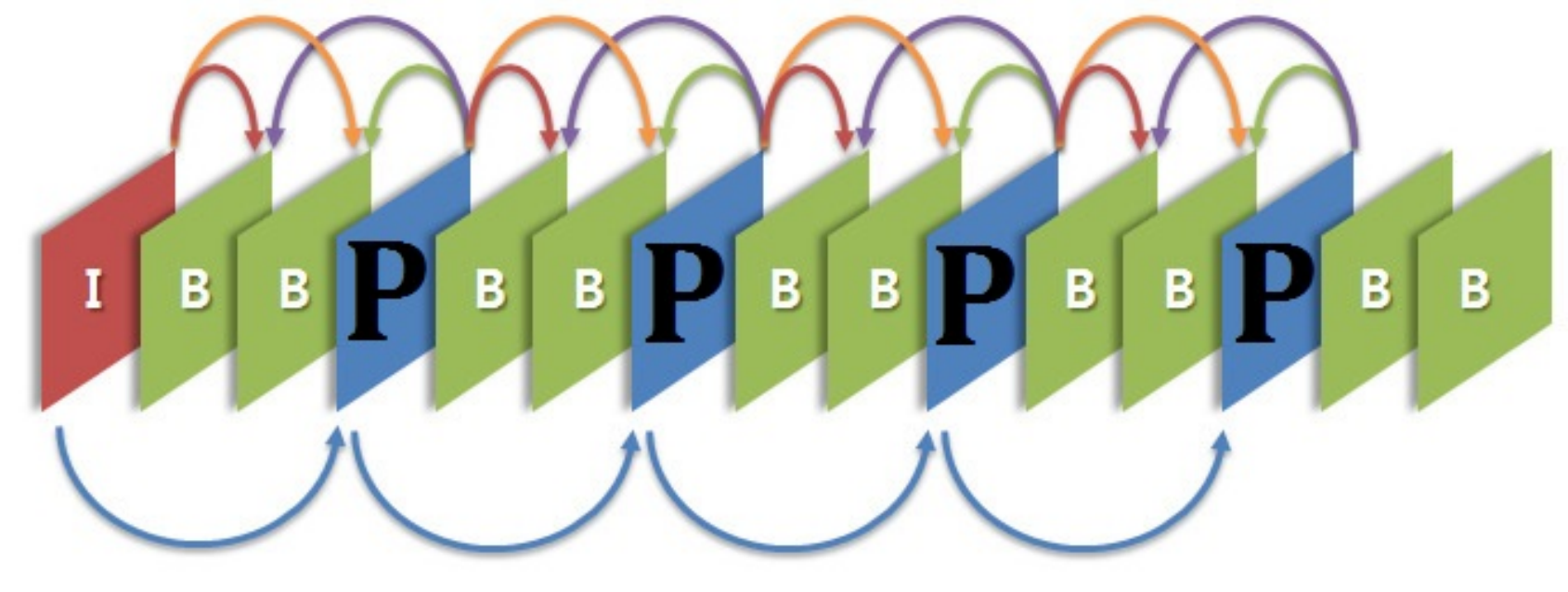
- I - Frame
- P - Frame
- B - Frame

MPEG Compression - Iframe



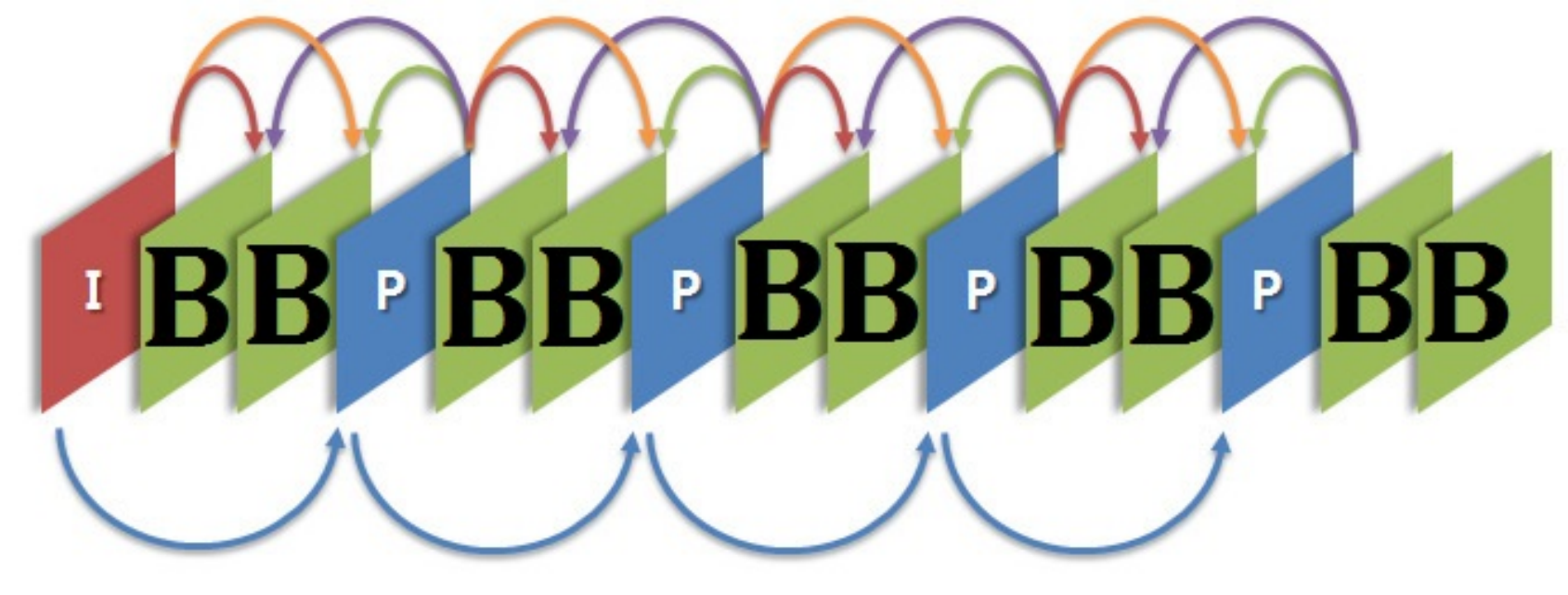
- *Intra Coded Frame*
- video key-frame
- all information stored internally
- compressed using JPEG

MPEG Compression - Pframe



- *Predictive Coded Frame*
- video reference-frame
- information only partially stored internally
- external information from previous P-/I-frames

MPEG Compression - Bframe



- *Bidirectional Coded Frame*
- minimal size video frame
- as few internal information as possible
- external information from previous as well as subsequent P-/I-frames

Did you encounter results of this type of compression before ?

Probably

Use of the previous I-frame

substitute grey I-frame generation

Table 9: Results of a missing I-frame

Sources

Imagesources

- original color space image:
  http://www.cambridgeincolour.com/tutorials/bit-depth.htm
- fps comparison gif: https://thumbs.gfycat.com/
- interlacing images: https://goo.gl/Sei9qj
- chroma subsampling:
  https://www.cinema5d.com/chroma-subsampling/
- discrete cosine transformation & quantization:
  https://upload.wikimedia.org/wikipedia/commons/2/24/DCT-8x8.png
  https://abyx.be/post.php?id=12
  http://imagebank.osa.org/getImage.xqy?img=QC5sYXJnZSxhby0zNy0yNi02MjEzLWcwMDE
- MPEG compression:
  http://yearningsoul.tistory.com/entry/GOP-Group-of-Pictures
- Missing I-Frame:
  https://www.youtube.com/watch?v=qbGQBT2Vwvc

Informationsources

- http://www.eetimes.com/document.asp?doc_id=1275437
- https://commons.wikimedia.org/w/index.php?curid=12624397
- http://electronics.howstuffworks.com/dtv4.htm
- https://www.youtube.com/watch?v=buSaywCF6E8
- SEVERAL wikipedia pages:
  - https://de.wikipedia.org/wiki/H.264
  - https://en.wikipedia.org/wiki/Chroma_subsampling
  - https://en.wikipedia.org/wiki/Interlaced_video
  - https://de.wikipedia.org/wiki/P-Frame
  - https://en.m.wikipedia.org/wiki/Quantization_(image_processing)
  - https://en.wikipedia.org/wiki/Huffman_coding
  - etc.

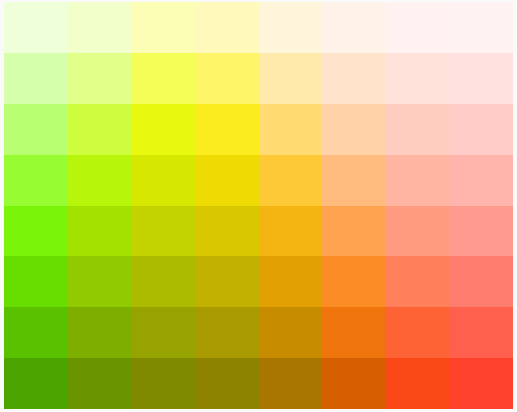
Computer Programs for Image changes

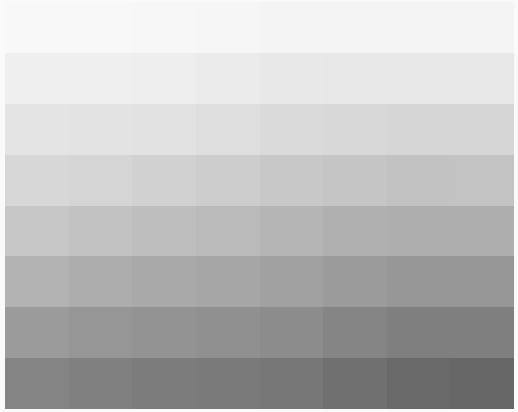
- reaConverter 7 Standard:
  https://www.reaconverter.com/
- Microsoft Paint
- .gif to .mp4 converter:
  http://www.online-convert.com/
- .mp4 to .png converter:
  http://de.office-converter.com/MP4-to-PNG
- VLC media player (.mp4 to .png):
  http://www.vlc.de/
- Snipping Tool:
  http://www.chip.de/downloads/Microsoft-Snip_82591670.html

- Overleaf LaTeXeditor
  https://www.overleaf.com/dash
  - this presentation:
    https://www.overleaf.com/9740717cwthrwncbzrk#/35526544/

Questions?

Additional - Discrete Cosine Transformation





248	248	247	246	244	244	244	244
239	239	238	235	232	231	231	231
228	227	226	222	218	216	214	214
215	213	209	205	200	197	194	195
199	194	190	186	181	176	174	174
179	173	169	166	161	155	151	151
155	150	147	144	140	133	127	127
132	128	124	122	119	112	105	103

$$g = \begin{bmatrix} -76 & -73 & -67 & -62 & -58 & -67 & -64 & -55 \\ -65 & -69 & -73 & -38 & -19 & -43 & -59 & -56 \\ -66 & -69 & -60 & -15 & 16 & -24 & -62 & -55 \\ -65 & -70 & -57 & -6 & 26 & -22 & -58 & -59 \\ -61 & -67 & -60 & -24 & -2 & -40 & -60 & -58 \\ -49 & -63 & -68 & -58 & -51 & -60 & -70 & -53 \\ -43 & -57 & -64 & -69 & -73 & -67 & -63 & -45 \\ -41 & -49 & -59 & -60 & -63 & -52 & -50 & -34 \end{bmatrix}$$

$$G = \begin{bmatrix} -415.38 & -30.19 & -61.20 & 27.24 & 56.12 & -20.10 & -2.39 & 0.46 \\ 4.47 & -21.86 & -60.76 & 10.25 & 13.15 & -7.09 & -8.54 & 4.88 \\ -46.83 & 7.37 & 77.13 & -24.56 & -28.91 & 9.93 & 5.42 & -5.65 \\ -48.53 & 12.07 & 34.10 & -14.76 & -10.24 & 6.30 & 1.83 & 1.95 \\ 12.12 & -6.55 & -13.20 & -3.95 & -1.87 & 1.75 & -2.79 & 3.14 \\ -7.73 & 2.91 & 2.38 & -5.94 & -2.38 & 0.94 & 4.30 & 1.85 \\ -1.03 & 0.18 & 0.42 & -2.42 & -0.88 & -3.02 & 4.12 & -0.66 \\ -0.17 & 0.14 & -1.07 & -4.19 & -1.17 & -0.10 & 0.50 & 1.68 \end{bmatrix}$$

$$G_{k,l} = \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} g_{n,m} \cos\left(\frac{(2n+1)(2k+1)\pi}{4N}\right) \cos\left(\frac{(2n+1)(2l+1)\pi}{4M}\right)$$