Samenvatting DMA

Introduction

New emerging stuff

- Computational Imaging
  - New Images are made from a set of images, e.g. new angles
  - Digital computations are performed on the captured signals before rendering them to the output
- Light field images
  - SMoE: Special cameras that capture information about the direction of light within the picture and be used to change the focus AFTER taking pictures
  - NERF: Multiple images + light fields can be used to create a 3D space
    - Very computationally intensive
    - Promising, getting more optimized
  - 3D Gaussian Splatting: Similar to SMoE
    - + clustering of colors through probability
    - + alpha compositing (= blending of multiple images) = splatting

Chapter 8
Coding moving pictures: motion prediction

Recap

- Standards define bitstream format and DEcoding process
- Significant temporal correlation in natural image sequences
  - Up to 8 frames
  - Exploited in compression models by applying motion compensated prediction before transformation

8.1 Exploiting temporal redundancy

Challenges

- Motion Estimation (ME) yields a residual signal
  - Transform coding is less efficient
- Motion can be complex
  - Accuracy vs search complexity
- Lighting variation in time
- Object (dis)appearance
- Makes encoder and decoder a lot more complex
- Motion information needs to be coded as well

Projected motion = true 3D motion projected on 2D plane
Apparent motion = variations in signal intensity between frames

ME is not about estimating true motion

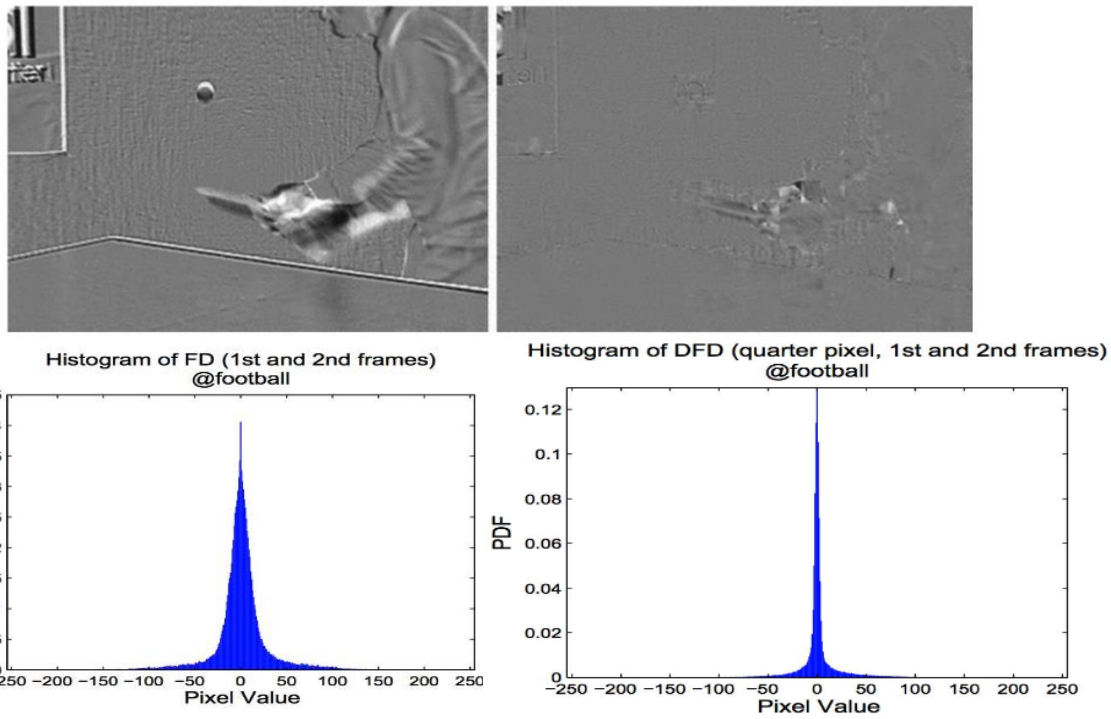
- Minimizing bits to code the residual signal
- Most commonly using block based 2D translational model

2 main approaches to form prediction

- Frame differencing
  - Pixel-per-pixel prediction between two frames
  - Prediction error = Frame Difference (FD)
    - Difference between co-located pixel values

- Motion-compensated prediction (MCP)
  - ME using a motion model
  - Motion Compensation (MC): modify reference frame content according to motion information from ME
  - Result is MCP or displaced frame (DF)
  - Prediction error = displaced frame difference (DFD)

FD vs. MCP



Effect of MC on correlation between current frame and reference frame

- Higher correlation
- Lower-energy residual to be coded
- = better compression

Three classes of prediction

- Forward
- Backward
- Bi-directional

Assuming perfect (forward) prediction, we have $S_k[\mathbf{p}] = S_{k-j}[\mathbf{p}-\mathbf{d}]$

- pure translational motion model (at pixel level)
- $S_k[\mathbf{p}]$ pixel at location $\mathbf{p}$ = [x,y] in frame at time *k*
- $\mathbf{d}$ = motion vector

ME is an ill-posed problem =

- Not necessarily unique solution
- Maybe no solution at all
- Sensitive to noise

Key issues in motion estimation

- motion model
  - complex vs. simple
- matching criterion
  - how 'good' is a candidate match?
- region of support
  - set of pixels to which the motion model is applied
- search range
  - how 'far' to look for potential matches
- search strategy
  - execution with above choices + reduce computational complexity of exhaustive search

= TRADEOFF BETWEEN ACCURACY AND OVERHEAD

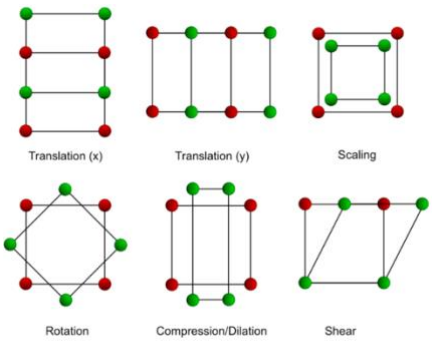
8.2 Motion models and motion estimation

Three causes for apparent motion

- Global motion (camera)
- Local motion of object
- Illumination change

Motion model examples

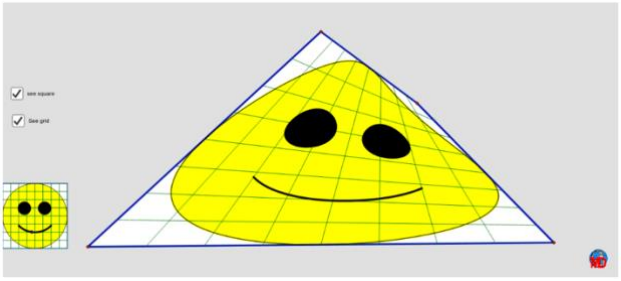
- Affine



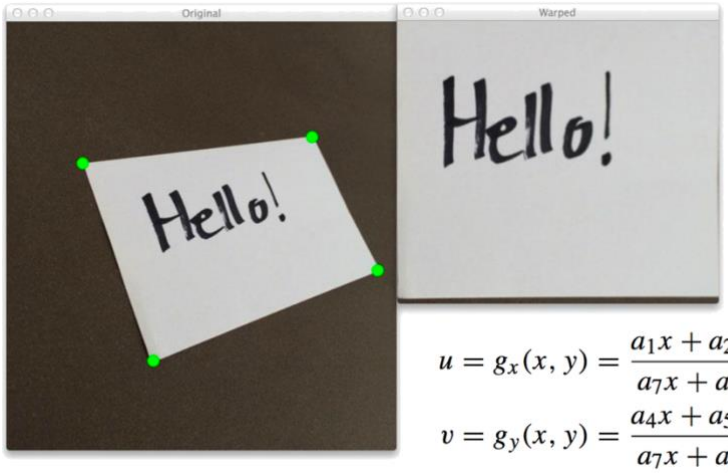
$$u = g_x(x, y) = a_1x + a_2y + a_3$$
$$v = g_y(x, y) = a_4x + a_5y + a_6$$

- Bilinear

$$u = g_x(x, y) = a_1xy + a_2x + a_3y + a_4$$
$$v = g_y(x, y) = a_5xy + a_6x + a_7y + a_8$$



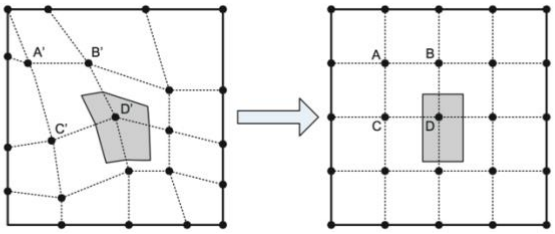
- Perspective



$$u = g_x(x, y) = \frac{a_1x + a_2y + a_3}{a_7x + a_8y + 1}$$
$$v = g_y(x, y) = \frac{a_4x + a_5y + a_6}{a_7x + a_8y + 1}$$

Example: node- or mesh-based warping

- estimate motion of nodes (e.g., using block-based matching)
- calculate model parameters a_i
- apply motion model to all pixels of each patch
- difference becomes the MCP



Translation only models

- Widely used because of performance and simplicity
- Why? Motion is rarely purely translational??
  - Most motion is approximately translational!
- Is a subset of the perspective model $u = g_x(x, y) = x + a_1 = x + d_x$
  $v = g_y(x, y) = y + a_2 = y + d_y$

Translation-only vs warping

- Warping has better prediction for basic BBME
- Warping models up to 50 times more complex though

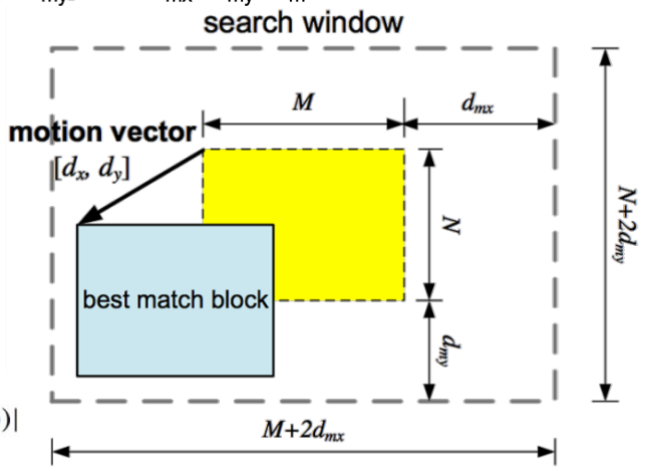
8.3 Block matching motion estimation (BMME)

= Translational block matching

How far to look for best match block?

In practice, search grid is restricted to $[\pm d_{mx}, \pm d_{my}]$ with $d_{mx}=d_{my}=d_m$

- $15 \leq d_m \leq 63$



Matching criteria

- Assessed with some form of block distortion measure (BDM)
- Motion vector is then then the motion pair (i,j) with lowest BDM
- Many BDMs exist e.g. SAD, SSD

$$\mathrm{SAD}(i,j) = \sum_{x=0}^{N-1}\sum_{y=0}^{M-1} |(s(x,y) - s_r(x+i, y+j))|$$

$$\mathrm{SSD}(i,j) = \sum_{x=0}^{N-1}\sum_{y=0}^{M-1} (s(x,y) - s_r(x+i, y+j))^2$$

- A transform e.g. SATD (Sum of absolute transformed differences) can be added prior to computing SAD
  - To factor in an estimate of the coding cost (like regularization in AI)

$$\mathrm{SATD} = \sum_{n=1}^{N-1}\sum_{m=0}^{M-1} |r_H(n,m)| \qquad \mathbf{R}_\mathrm{H} = \mathbf{H}.(\mathbf{S}_1 - \mathbf{S}_2).\mathbf{H}^\mathrm{T}$$

  - **H** is a Hadamard matrix
  - **R** is the (Hadamard) transform of the difference between blocks $\mathbf{S}_1$ and $\mathbf{S}_2$

- Issues
  - May have multiple local minima
  - Large variation of BDM values, even within a single block

Properties of block motion fields and error surfaces

- Block motion field is generally smooth and slowly varying
  - High correlation with motion of neighbor blocks
- Block size has a big effect on overhead. (number of bits)
  - Not visible if you only look at PSNR

Effect of search range (d_m) in pixels

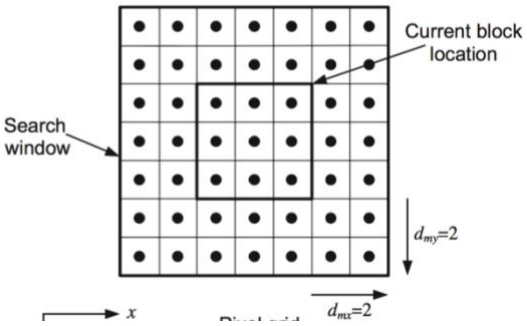
- Complexity increases

- Small range might not capture fast moving objects = motion failure
- Range should be dictated by content type AND video format
    - Higher motion + higher resolution

Motion failure

- DFD (displaced frame difference) has significant artifacts
- Transform performs poorly
- Solutions
    - Intra vs inter
    - Sub pixel ME
    - In-loop deblocking filter (ch9-10)

8.4 Reduced complexity motion estimation



Complexity of full search

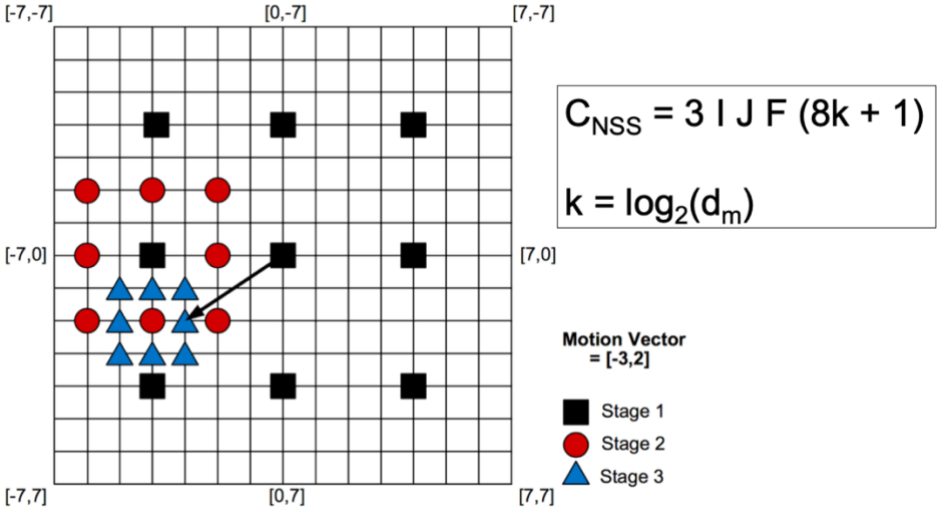
- Per block: (2*dm+1)^2 candidates, m*n pixels per MV (motion vector), 3 operations per pixel

$$\Rightarrow C_{FS} = 3\ I\ J\ F\ (2d_m + 1)^2$$ (I x J frame resolution, F fps)

Approaches to reduce complexity

- Reduced complexity BDM
    - SAD, or don't evaluate all pixels (depending on activity)
- Sub-sampled block motion field
    - ME on a subset of blocks
- Hierarchical search techniques
- Reduced set of MV candidates **(!!!)**
- Intelligence initialization and termination **(!!!)**

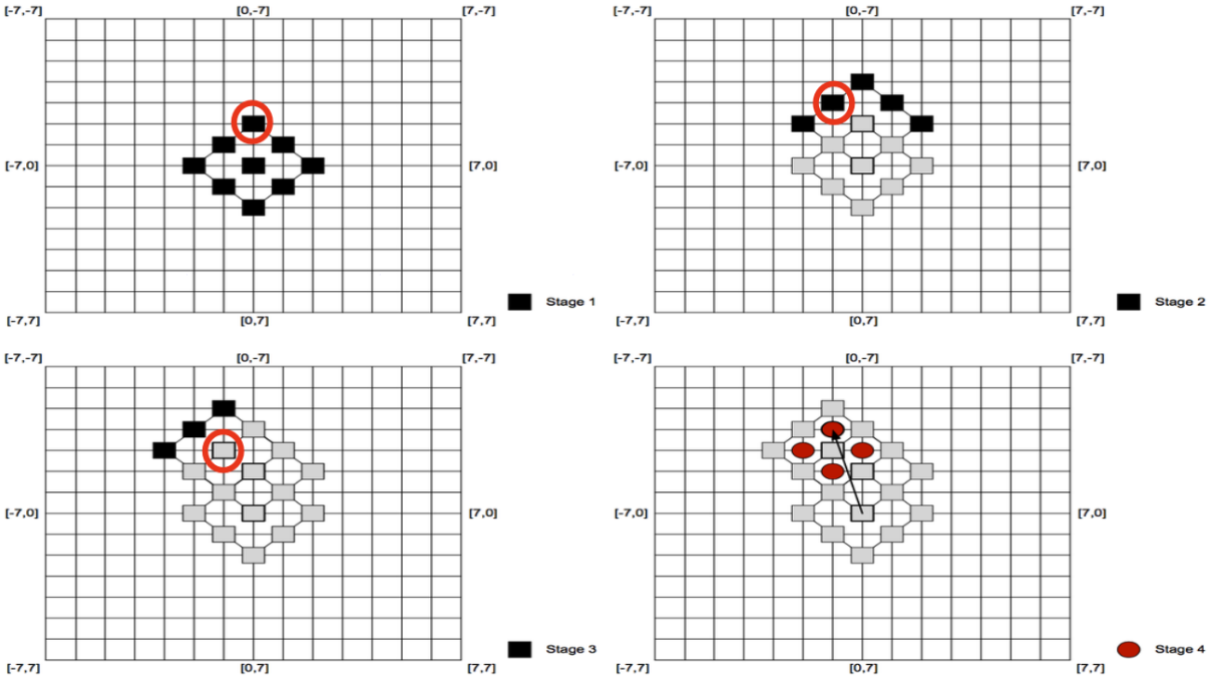
Example: N-step Search (NSS)



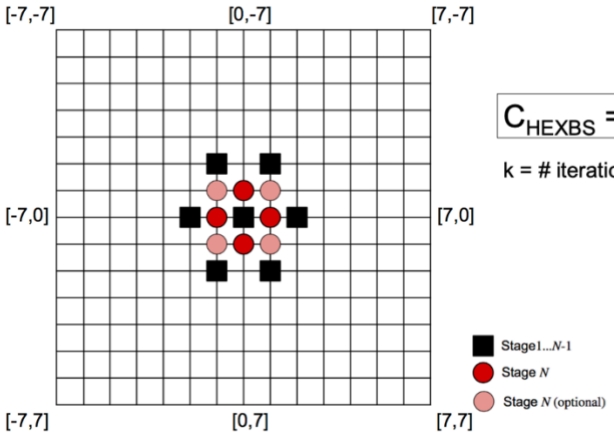
$$C_{NSS} = 3\ I\ J\ F\ (8k + 1)$$

$$k = \log_2(d_m)$$

Diamond Search (DS)



Hexagonal search (HEXBS)



$$C_{HEXBS} = 3\ I\ J\ F\ (7 + 3k + 4)$$

k = # iterations to reach minimum

Reduced complexity ME might get trapped in a local minimum!!!

- Need better starting point than [0,0] = initialization
- Possible starting locations include

$$\hat{\mathbf{d}}_P = \{[0,0], \mathbf{d}_A, \mathbf{d}_B, \mathbf{d}_C, \mathbf{d}_D, \text{med}(\mathbf{d}_A, \mathbf{d}_C, \mathbf{d}_D), \text{med}(\mathbf{d}_A, \mathbf{d}_B, \mathbf{d}_C)\}$$

B	*C*	*D*	
A	***P***		

- Early termination methods also significantly speed-up BMME

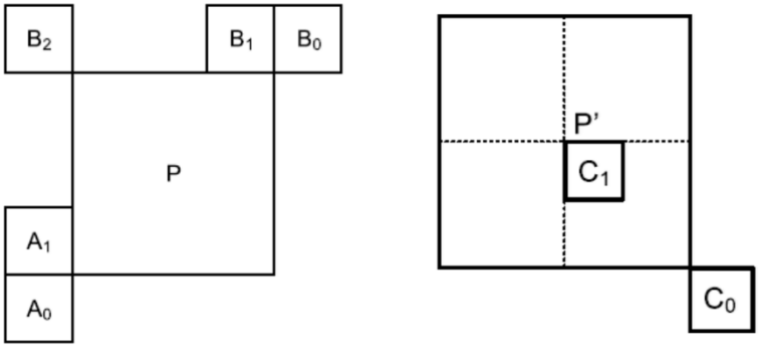
8.5 Skip and merge modes

Skip mode

- Special inter-prediction mode
- Motion information current block is predicted from neighbors, but *without prediction residual*
    - E.g. content with consistent global motion
- Low bit cost
- Reduces overall computational complexity

Merge mode

- Extension of skip mode
- To improve coding performance for content with homogeneous motion
- List of candidate MVs
- No prediction residual
- If one of these candidates has best RD (Rate-Distortion), additional bits are used to signal the spatio-temporal index
    - Left = spatial merge candidates, Right = temporal merge candidates



8.6 Motion vector coding

Motion coding overhead increased drastically with new codecs!

Motion Vector Prediction (MVP)

- Exploit smoothness of motion field = spatial correlation

Entropy coding of motion vectors

- MVP results in decorrelated residual signal
    - Traditional entropy coding techniques can be applied

Chapter 9
Hybrid Block-based Video Codec

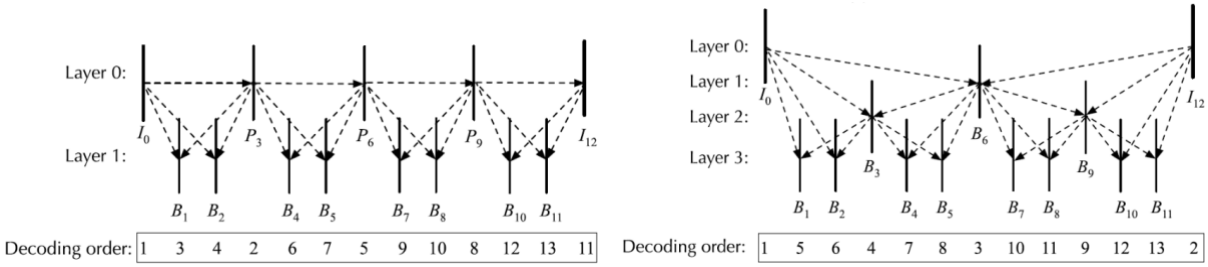
9.1 Block-based hybrid model for video compression

How to integrate motion estimation (ch8) into a practical video compression framework?

Hybrid structure combining
- Block-based motion prediction = ME
- Block-based transform coding = compression

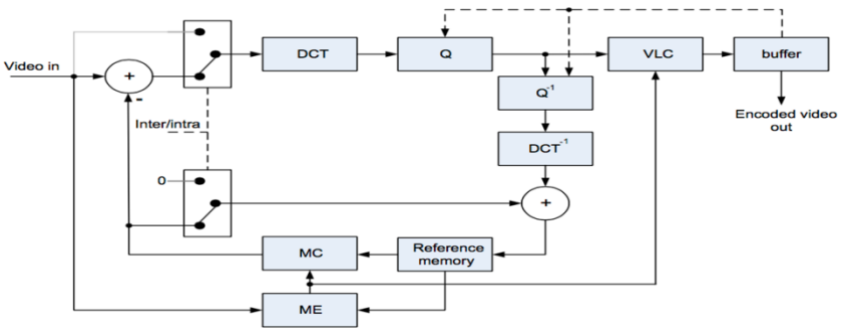
Different types of frames
- I-frames: INTRA coded
- P-frames: INTER coded, based on 1 reference frame
- B-frames: INTER coded, based on 1 or 2 reference frames
- IDR frames

Many different structures are possible, e.g. MPEG-2 left, H.264/AVC right

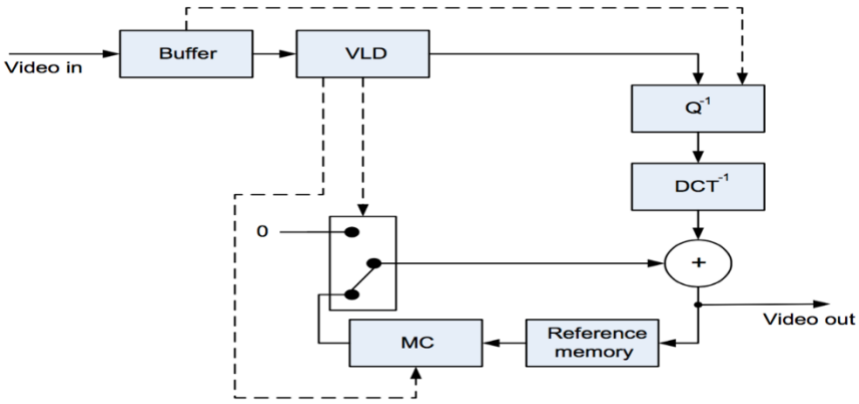


Inter vs Intra coded frames
- Inter = P- and B-frames
  - Compression takes place across multiple frames, where the encoding scheme only keeps the information that changes between frames
  - Loopback in the video encoding scheme
- Intra = I-frame
  - All the compression is done within that single frame
  - Straight line in the encoding scheme: DCT – Q – VLC (Variable Length Coding) – buffer – encoded video out

Scheme: INTER frame encoding



Decoder



Many coding tools are aimed at improving the characteristics of DFD signal (smoothing)

- Because autocorrelation of DFD is low so coding gain of decorrelating transforms like DCT, KLT will be smaller when applied to DFD signals

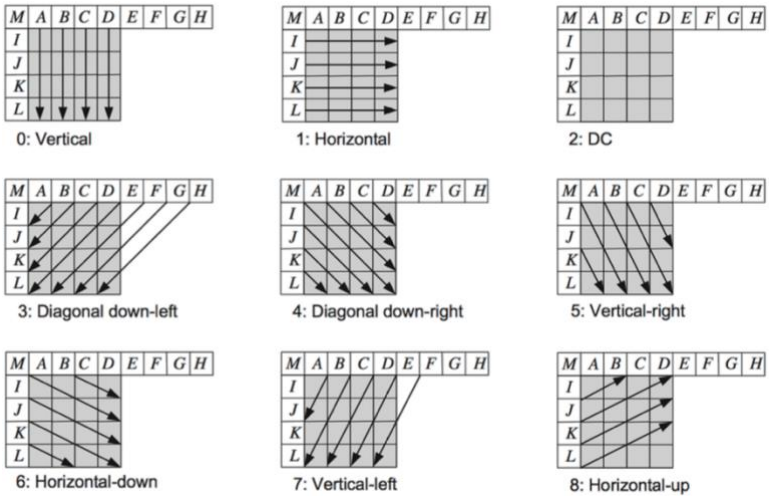
9.2 Intra-frame prediction

Prediction in the spatial domain, before transform coding

- Pixel blocks are predicted based on extrapolation of surround pixel values
- H.264/AVC
  - 4x4 and 16x16 blocks (luma)
  - 8x8 (High profiles)

Intra prediction in H.264/AVC

- 4 modes for 16x16
- 9 modes for 4x4 and 8x8
  - 1 DC mode (avg the values) and 8 directional modes



M	A	B	C	D	E	F	G	H
I	a	b	c	d				
J	e	f	g	h				
K	i	j	k	l				
L	m	n	o	p				

Mode 0 (Vertical)

$$\{a, e, i, m\} = A$$

Mode 1 (Horizontal)

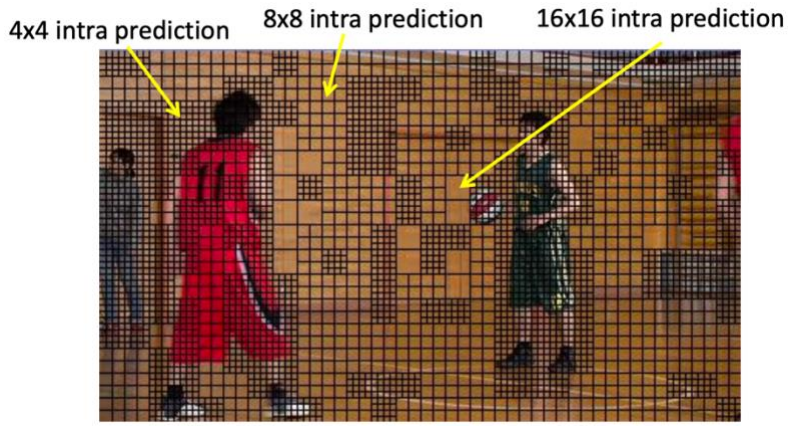
$$\{a, b, c, d\} = I$$

Mode 2 (DC)

$$\{a, b, \cdots, p\} = \left\lfloor \left(\frac{A + B + C + D + I + J + K + L}{8} \right) + 0.5 \right\rfloor$$

Mode 4 (Down-right)

$$\{a, f, k, p\} = \left\lfloor \left(\frac{A + 2M + I}{4} \right) + 0.5 \right\rfloor$$



9.3 Sub-pixel motion estimation

More accurate matches might be found with sub-pixel displacements

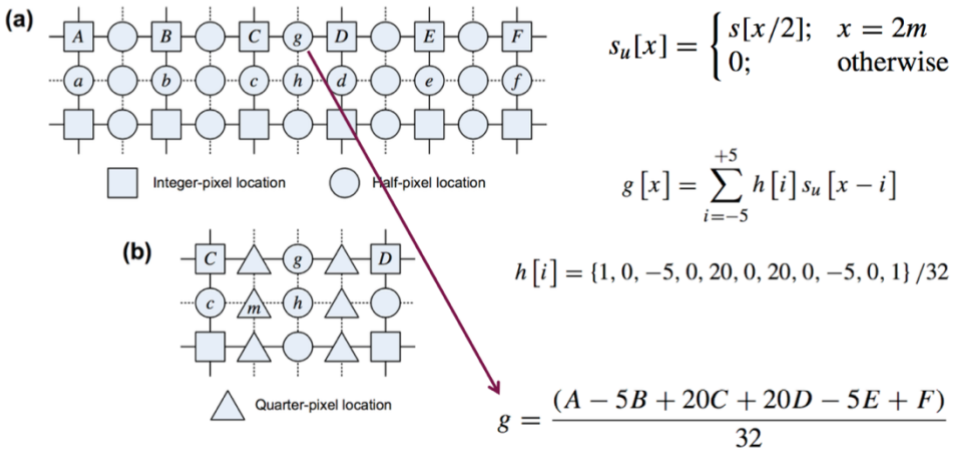
- Gains up to 2dB when going to 1/8 pixel accuracy = higher PSNR

First: what is interpolation? = Generate additional pixel values at non-integer coordinates.

Various possible approaches

- Interpolate current block + search window, then ME
- Interpolate only search window, then ME
- Integer-pixel ME, then refine with interpolated block + search window
- Integer-pixel ME, then refine with interpolated search window

Interpolation method in H.264/AVC



$$s_u[x] = \begin{cases} s[x/2]; & x = 2m \\ 0; & \text{otherwise} \end{cases}$$

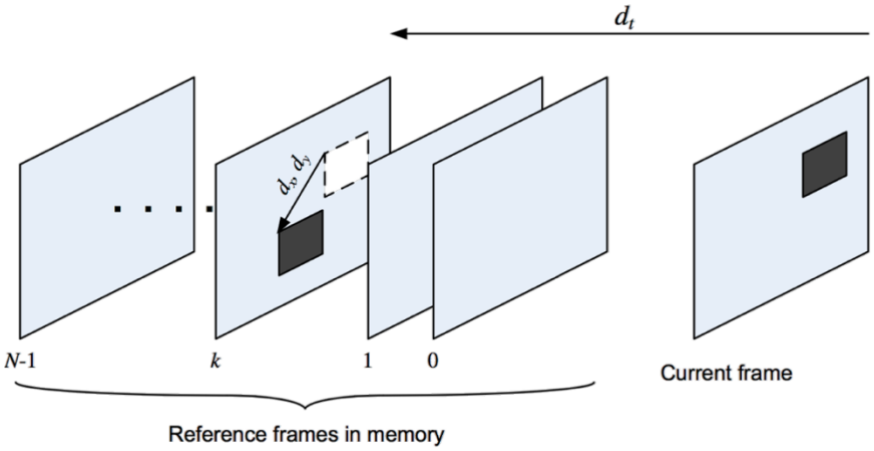
$$g[x] = \sum_{i=-5}^{+5} h[i]\, s_u[x-i]$$

$$h[i] = \{1, 0, -5, 0, 20, 0, 20, 0, -5, 0, 1\}/32$$

$$g = \frac{(A - 5B + 20C + 20D - 5E + F)}{32}$$

9.4 Multiple reference frame ME

Used to exploit long term statistical dependencies in video sequences



Lager memory size also means longer searching time, how to fix?

⇨ Hierarchical B prediction enables temporal scalability

9.5 Variable block sizes for ME

Variable block size has additional benefits

- Shape and size optimized in rate-distortion sense
- Constant balance between prediction accuracy and coding overhead

9.6 Variable sized transforms

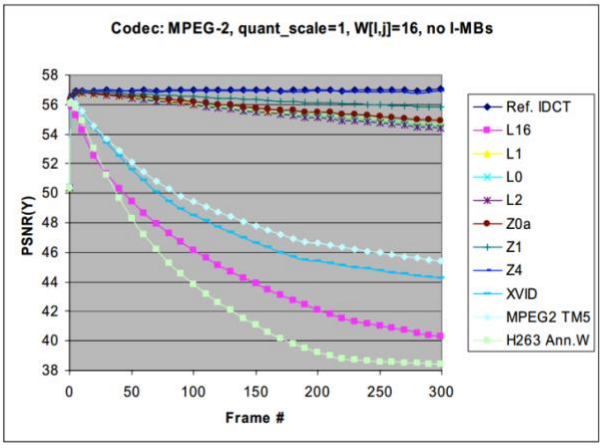
Match transform size to prediction block?

In recent standards a smaller-sized transform (4x4) is chosen

- Corresponding to smallest size of prediction block
- Transform operates on decorrelated signals, requiring less decorrelating performance
- Smaller transforms lead to less quantization noise (ringing)
- To accommodate smooth areas, DC coefficients are further transformed

In early codecs, implementation of DCT (Discrete Cosine Transform) and IDCT algorithms was considered a major technical challenge

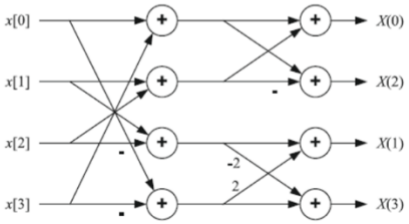
- That's why the JPEG, MPEG and H.261 standards did not dictate specific algorithms, but only defined precision requirements.
- This made it possible for manufacturers to implement the algorithms with a certain freedom which made it more optimized
- Drawback: Lack of standardization makes it impossible for exact decoding of MPEG-encoded videos across different decoders



- L_X: low-precision IDCT approximations
- Z_Y: higher-accuracy IDCT approximations

Integer transform in H.264/AVC – See slides!

- Conclusion: Efficient architecture (in 1D)
  - only eight additions/subtractions and two 1-bit shifts



9.7 In-loop deblocking operations

Blocking artifacts arise because of

- Motion estimation
- Transform coding of the difference signal

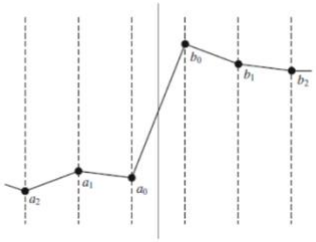
Determine whether

- Edge is part of image
- Edge is induced by coding => set thresholds depending on quantization

Filtered image is used for MC'ed prediction of future frames (in-loop!)

- Can improve compression performance
- Filtered image is often a more faithful reproduction of the original frame than a blocky, unfiltered image(!)

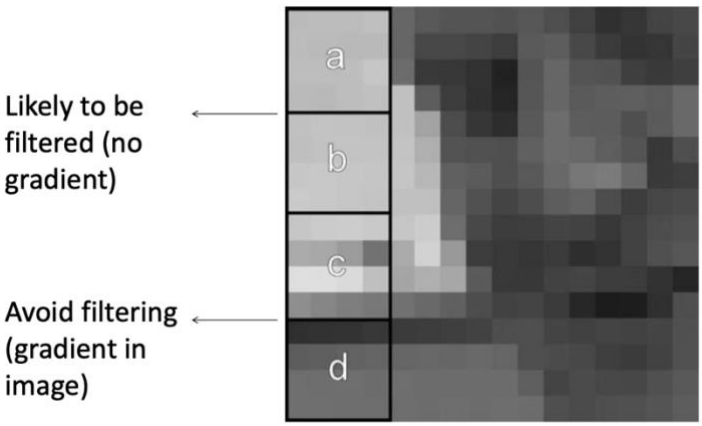
Example (1D):



1. Set thresholds $\alpha(QP)$ and $\beta(QP)$;
2. Filter a_0 and b_0 iff: $|a_0 - b_0| < \alpha(QP)$ AND $|a_1 - a_0| < \beta(QP)$ AND $|b_1 - b_0| < \beta(QP)$;

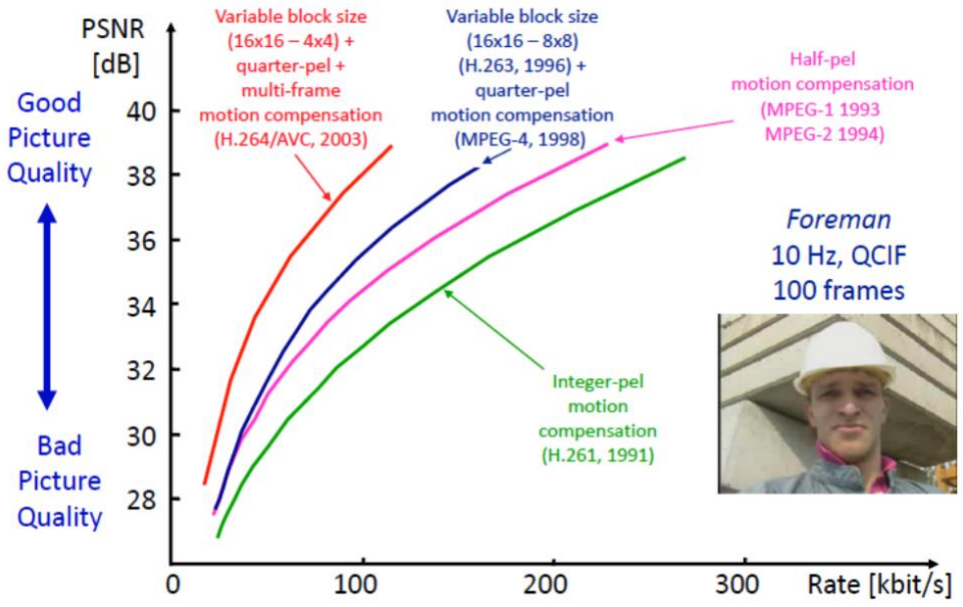
Switch off filter when significant change (gradient) across the block boundary is present in the original image

Example (2D):



Deblocking will give a visual result, where blocks disappear BUT also invisible results: compression gains due to higher correlation in the DFD signal

Performance Comparison



Chapter 10a
Measuring and Managing Picture Quality

10.1 General considerations

Motivations for visual quality assessment
- To compare different codecs across range of bit rates and content types
- To compare influence of parameters for a given codec
- To compare performance of different codecs across a range of channel impairments

Subjective assessment
- Requires many observers
- Closely controlled to ensure consistency and statistical significance
- Costly and time consuming but effective

Objective assessment
- Metrics that attempt to capture perceptual mechanisms of the HVS (Human Visual System)
- Issue: hard for simple metrics to assess quality also for PSNR!!
- Still mostly used MSE (mean square error)

10.2 Subjective testing

Widely used, 5 components:
1. Test material
2. Test conditions
3. Test subjects
4. Environment
5. Methodology

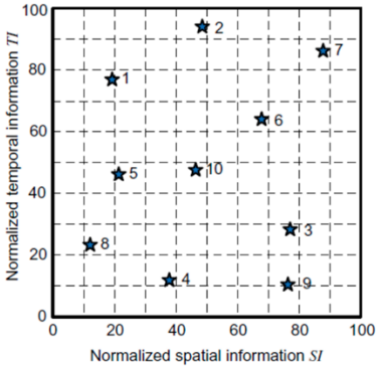
1.Test material
- Content may have large impact on rate-distortion performance
- Need for representative but challenging content
- Typically short sequences
  - P.910 approach
    - spatial information (SI) = standard deviation of frame Sz after Sobel filtering

$$\mathrm{SI} = \max_{\forall z} \left\{ \sigma_{\forall (x,y)} \left(\mathrm{Sobel} \left(\mathbf{S}_z(x, y) \right) \right) \right\}$$

    - temporal information (TI) = standard deviation of the difference, over consecutive frames, between co-located luma pixel values

$$\mathrm{TI} = \max_{\forall z} \left\{ \sigma_{\forall (x,y)} \left(\mathbf{S}_z(x, y) - \mathbf{S}_{z-1}(x, y) \right) \right\}$$

- Sobel filtering: Related to gradient of the image intensity
    - Black image with white contour lines, used for edge detection
- Activity/information coverage for a test set of sequences



2.Test condition

- Typical test: compare multiple codecs for range of sequences and coding rates
- Avoid excessive length of test session (viewer fatigue)
- Recency/order bias of the videos
- Fatigue bias: less reliable judgements near the end of the session

3.Test subjects

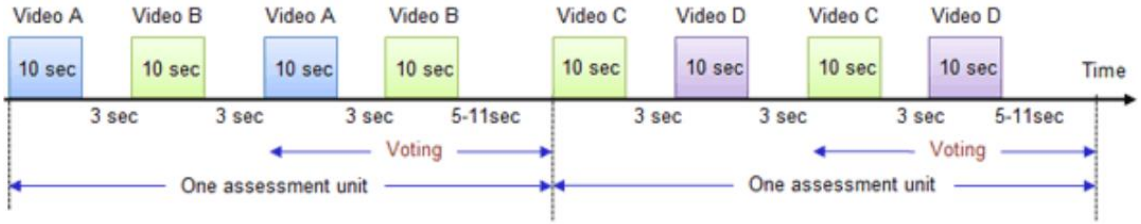
- Expert or non expert
    - Majority should be non expert
- Expertise bias
- Check eyes of participants

4.Environment

- Laboratory vs home
- Parameters: display, ambient lighting, viewing distance, audio quality

5.Methodology

- Bringing it all together: duration, prepare participants, video order, grading, recording test conditions
- Double Stimulus Continuous Quality Scale (DSCQS)
    - Assess how well the test performs compared to signal
        - Good for cases where qualities are similar
    - Arranged as sequence of paired clips (A and B)



- Double Stimulus Impairment Scale (DSIS)
    - Similar to DSCQS, except that
        - Pair is presented only once
        - Assessor knows which clip is the original
- Pair comparison method
    - Similar to double stimulus but both systems are shown at once!

- Single stimulus methods (ACR) absolute category rating
  - Assessor is presented with test conditions + original as hidden reference
  - Advantage: reduced testing time
  - Disadvantage: lower discrimination power

Bias

- Anchoring bias: influence by the first video they view, which serves as anchor to subsequent judgements
- Recency bias
- Halo effect: Rating based on one single pos/neg aspect
- Central tendency bias: People avoid extreme response categories
- Acquiescence bias: agree with statements to please / disagree to avoid making mistakes
- Response bias: faking good/bad response

	Subjective	Objective
Cost	High	Low
Discrimination power	Small discrimination power	Large discrimination power, i.e. small changes can be detected
Scope	Universally accepted results	Results only applicable within boundaries , i.e. PSNR does not work with packet loss
Usage	Only to compare entire systems, RD evaluations are not feasible	To compare entire systems, but also during the RD process

Result

- Statistical analysis and processing
  - Mean Opinion Score (MOS)
    - mean score across all observers and repetitions

$$MOS_{cs} = \bar{u}_{cs} = \frac{1}{R}\frac{1}{K}\sum_{r}\sum_{k} u_{kcsr}$$

    - 5: Excellent, 1: Bad
  - Difference of MOS (DMOS)
    - If Hidden Reference is used

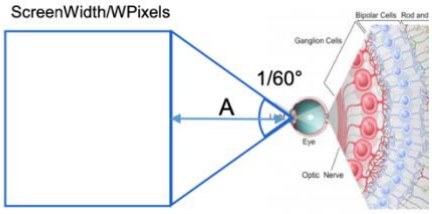
$$DMOS_{cs} = MOS_{REF,s} - MOS_{cs}$$

    - 0: Excellent, 4: Bad

Observer	k
Sequence	s
Condition	c
Repetition	r

- Use of confidence intervals
- Reject outliers

The eye can differentiate 1 bow minute = 1/60 degree
How to calculate the distance you have to be from the screen?



$$\tan((1/60)/2) = (\text{ScreenWidth/WPixels}/2)/A$$

$$A = (\text{ScreenWidth/WPixels}/2)/\tan((1/60)/2)$$

$$A \approx \text{ScreenWidth} * (3438/\text{WPixels})$$

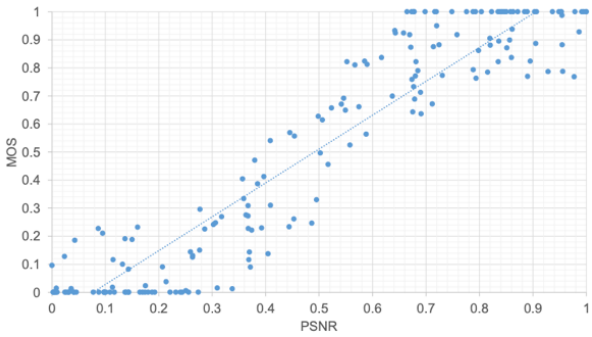
4K: $A \approx 0.9$ ScreenWidth
HD: $A \approx 1.8$ ScreenWidth

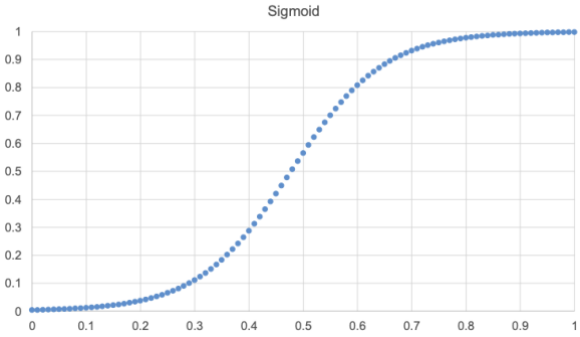
10.3 Test data sets and how to use them

Relationship to subjective score (MOS) and objective metric

- Not linear
- Translate objective metric to MOS scale
  - Normalize MOS values (1->5 => 0->1)
  - Model using sigmoid function
  - Dm and G are constants to be trained on the data
- Get to know how PSNR translates to MOS scale (excellent, good, avg, poor, bad)

$$p_i = \frac{1}{1 + e^{(n_i - D_M)G}}$$





Evaluating metrics

- Linear correlation
  - Pearson Linear Correlation Coefficient (LCC)

$$r_p = \frac{\sum_{i=0}^{N-1}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=0}^{N-1}(x_i - \bar{x})^2}\sqrt{\sum_{i=0}^{N-1}(y_i - \bar{y})^2}}$$

- Rank-order correlation
  - Spearman Rank-Order Correlation Coefficient

$$r_s = \frac{\sum_{i=0}^{N-1}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=0}^{N-1}(x_i - \bar{x})^2}\sqrt{\sum_{i=0}^{N-1}(y_i - \bar{y})^2}}$$

- Outlier ratio
  - Outlier = a predicted data point that is greater than a threshold distance from the corresponding MOS point.

$$OR = \frac{K_{out}}{K} \qquad K_{out} = \sum_{i=1}^{K} k_i \qquad k_i = \begin{cases} 0 & \text{if } OM'(i) \in CI(i) \\ 1 & \text{otherwise.} \end{cases}$$

LCC = 1
SROCC = 1

0	0
0.1	0.05
0.2	0.1
0.3	0.15
0.4	0.2
0.5	0.25
0.6	0.3
0.7	0.35
0.8	0.4
0.9	0.45
1	0.5

LCC = 0.992
SROCC = 1

0	0
0.1	0.09
0.2	0.1
0.3	0.19
0.4	0.2
0.5	0.29
0.6	0.3
0.7	0.39
0.8	0.4
0.9	0.49
1	0.5

10.4 Objective quality metrics

Why use objective metrics?

- Algorithm improvement & benchmarking
- Rate-distortion optimization
- Streaming control

Types of metrics – Reference availability

- Full-reference (FR) metrics
  - Original material is available
  - E.g. PSNR or MSE
- No-reference (NR) metrics
  - No available reference material
  - Restricted to specific scenarios and distortion types like blur
- Reduced-reference (RR) metrics
  - Partial information about source for predicting the quality
  - Use features of original content and the distorted content

Reduced-reference metrics

- Example features
  - Edge information: Edge sharpness, strength and coherence are commonly used
  - Texture descriptors: Features such as texture energy, contrast and homogeneity used to evaluate similarity between original and distorted images

Types of metrics – Used information

- Pixel-based = use decoded pixels
- Stream-based = use video stream info without decoding
- Hybrid = both

PSNR [FR, Pixel]

- Consistent when comparing similar codecs based on the same test data

$$MSE = \frac{1}{w\,h}\sum_{i=0}^{w-1}\sum_{j=0}^{h-1}[I_{orig}(i,j) - I_{dist}(i,j)]^2$$

$$PSNR = 10.\log_{10}\left(\frac{{MAX_I}^2}{MSE}\right) \qquad MAX_I = 2^B - 1$$

- However, MSE will fail badly for certain impairments that in reality have little perceptual impact
- MSE assumes that signal quality is independent of all relationships + assumes that all samples contribute equally to the signal quality

- Also: texture masking is overlooked in PSNR

28 dB 28 dB

SSIM [FR, Pixel]

- Structural Similarity Image Metric
- Based on 'integrity of structural information' in an image
- Estimates the degradation of structural similarity based on statistical properties

$$\mathrm{SSIM} = \frac{(2\mu_s\mu_{s_R} + C_1)\,(2\sigma_{ss_R} + C_2)}{(\mu_s^2 + \mu_{s_R}^2 + C_1)\,(\sigma_s^2 + \sigma_{s_R}^2 + C_2)}$$

  - impaired image s and its reference s_R
  - C1, C2 constants to stabilize denominator
  - μ = local mean, σ = standard deviation, σ_{SSR} = covariance
  - applied on a sliding window, typically of size 11 × 11
- SSIM = Luminance x Contrast x Structure (!!!)
- Does better on texture masking than PSNR

0.95 0.81

VMAF [FR, pixel]

- Video Multimethod Assessment Fusion
  - State of the art
- Fusion of different quality metrics:
  - Visual Information Fidelity (VIF)
  - Detail Loss Metric (DLM)
  - Motion: Mean Co-Located Pixel Difference: measures temporal difference between frames on the luminance component

Other metrics

- VSNR (Visual Signal to Noise Ratio) = for still images
- VQM (Video Quality Metric) = combine measures of all types of noise

P.1201.2 [NR, Stream]

- Parametric non-intrusive assessment of audiovisual media streaming quality
- Estimates coding quality for IP-based video streaming applications

Chapter 10b
Rate Distortion & Rate Control

10.5 Rate-distortion optimization

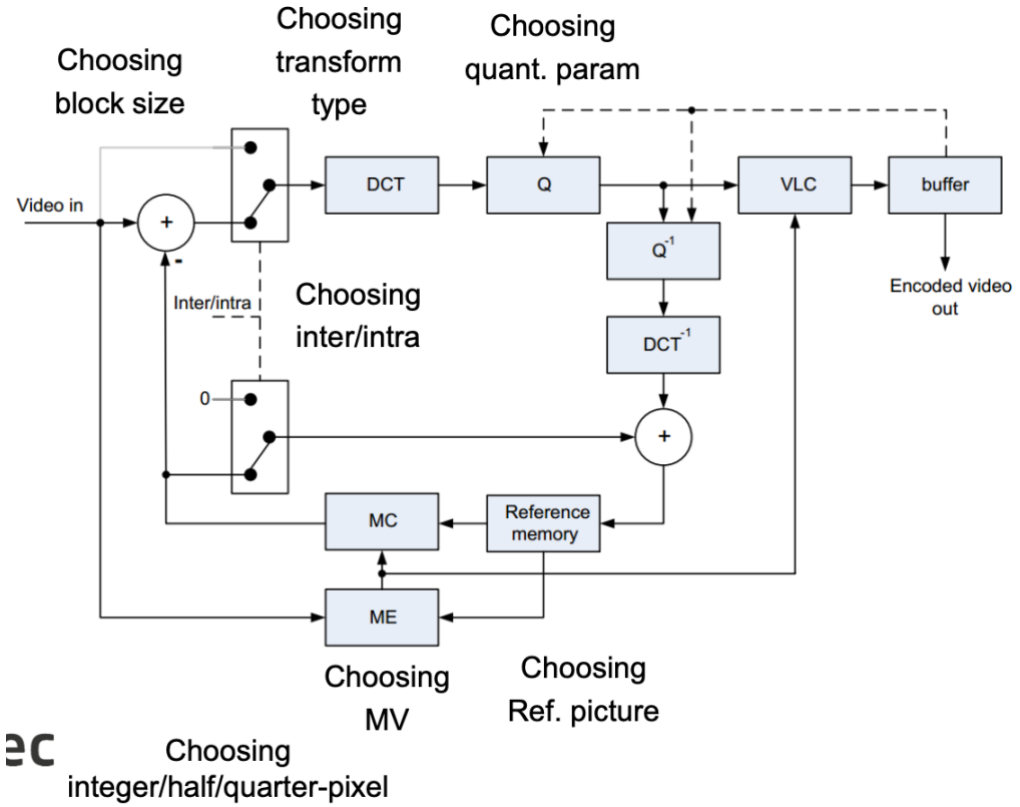
Bit rate depends on

- Video content: more movement = more bits
- Encoding algorithm
- Selected parameters: FPS, block size, encoder implementation,…

Classical RD theory:

- Shannon's separation principle = source and channel coding are treated independently
  - Compress as efficient as possible
  - Try to ensure error-free delivery
- Need for practical solutions
- Difficulties
  - Size of parameter space
  - Best settings vary for different spatio-temporal regions
  - RD costs are **not** independent for all coding units
    - Different MV at certain position influences all following blocks
    - …

Decisions during video encoding – a lot of choosing!



Lagrangian cost for coding unit i (quantization index j): $J_{ij}(\lambda) = D_{ij} + \lambda R_{ij}$

- R = rate, D = distortion

Minimizing J_{ij} gives

$$\frac{\partial(J_{ij}(\lambda))}{\partial R_{ij}} = \frac{\partial(D_{ij} + \lambda R_{ij})}{\partial R_{ij}} = \frac{\partial D_{ij}}{\partial R_{ij}} + \lambda_{\text{opt}} = 0$$

Optimum value of λ is given by the negative slope of the distortion function

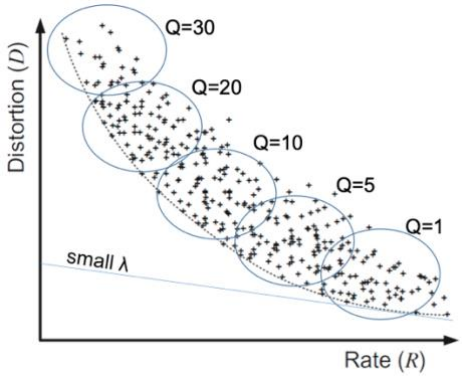
$$\lambda_{\text{opt}} = -\frac{\partial D_{ij}}{\partial R_{ij}}$$

- λ = 0 → minimize distortion
- λ = ∞ → minimize rate

However, slope is not known (requires R(D) or D(R) function)

Choose lambda according to Quantization Parameter (QP)

- Lower Q = finer quantization = better quality = higher bitrate
- E.g. small lambda for small Q



Impossible to do exhaustive RDO (R-D Optimization)

- Because of dependencies of encoding units

In practice

- Separate RDO evaluation for each coding unit
- Early termination strategies

Example for Intra prediction

- Q and lambda are given
- Parameter vector is varied over all possible modes for given coding unit
- Rate measured after entropy coding

10.6 Rate control

What causes bitrate to vary?

- Encoding Parameters: quantization, GOP structure (I, P, B frames with I higher bitrate)
- Source video complexity

Variable Bitrate (VBR)

- Keep distortion fixed at whatever bitrate cost
- Uniform quality distribution
- Size is unpredictable

Constant bitrate (CBR)

- Try to keep bitrate fixed at whatever distortion cost

VBR – CBR in applications

- DVD
  - CBR if we need to be sure that video will fit on DVD
  - Else, VBR
- Dynamic Adaptive Streaming over HTTP (DASH)
  - each 2-10s segment produced at equal bitrate
  - Variable bitrate within each segment

Rate Control – What is it?

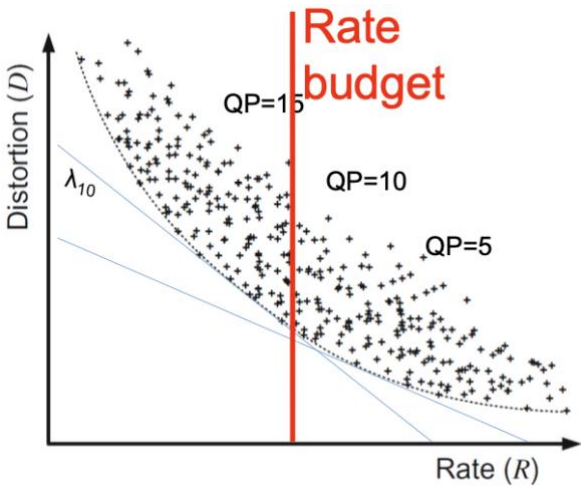
- Ensuring that video is delivered at rate, compatible with video complexity and channel capacity
- Without rate control, decoder buffers would under- and overflow resulting in playout jitter or loss

Difficulty of doing rate control and RDO jointly **(!!!)**

- To perform RDO, QP must be known
- To do rate control, QP depends on the content and thus on SAD or a variance measure
  - Actual SAD is only available after RDO is complete
    - Rate control estimates SAD of the coding unit
  - Header info like modes and MV are only known after RDO process is complete
    - Rate control needs to guess target bits for the current frame

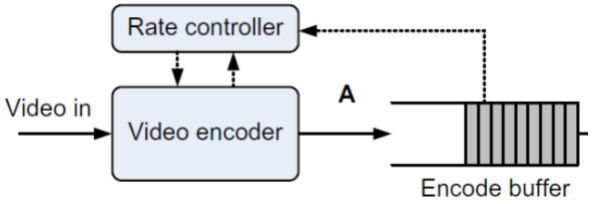
Example for rate-constrained motion estimation

- Impossible to eval all block sizes, MV,… for every Q
- Solution: 'Hypothetical reference decoder' (HRD)



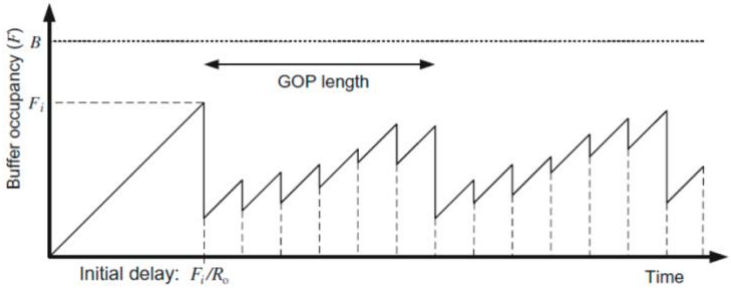
HRD

- Used in H.264/AVD, HEVC,…
- Buffer: leaky bucket approach
  - Constant rate flow to and from the channel
- Contains rate-quantization model that dynamically measures buffer level and content complexity
  - Sends QP to the encoder



Rate variability and delay

- Output rate R0
- Initial occupancy Fi
- Buffer size B



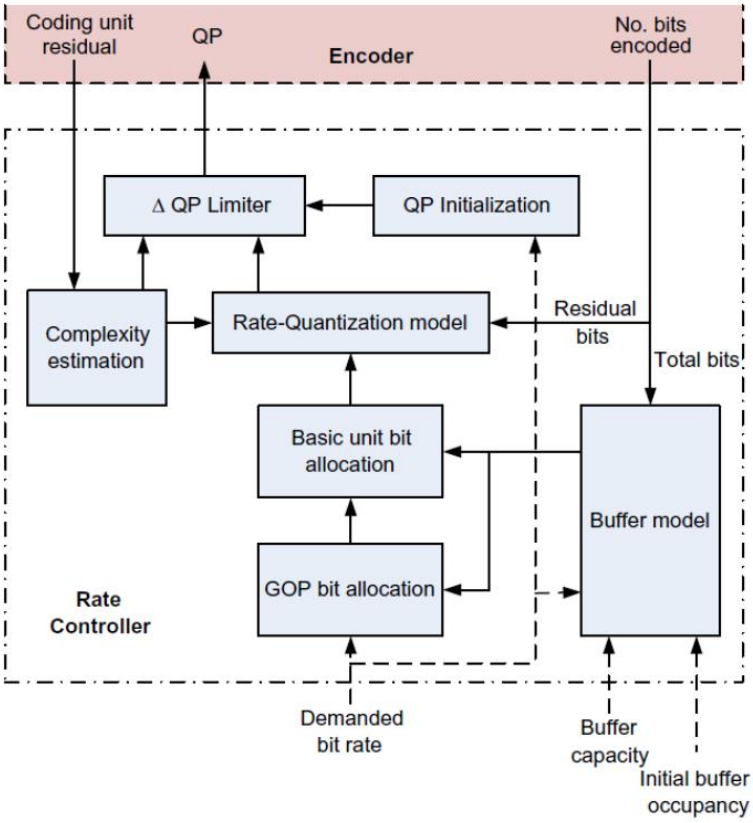
Variability of bitrate determines delay:

- Channel has capacity for 40kB frames @ 25fps
- Largest frame: 1MB = 1second delay before decoding can start

In practice

- Complexity estimation
  - Estimate complexity of residual
- Rate-quantization model
  - Describes how available bits are related to Q values

!!!!



2 pass encoding

- Pass 1:
  - Get an idea of how to distribute bits in the video
    - Fix QP so quality is distributed uniformly
    - Size is unpredictable
- Pass 2:
  - Use pass 1 to guide rate control
  - To end up at the exact amount of bits needed

Artefacts

- Pumping
  - By introducing a QP difference between frames the PSNR average can be improved
  - This introduces extra bit cost for the first frame of a GOP, afterwards better predictions follow
  - Extreme VBR and quality: I and P become more costly. Less bits on B and b
  - Extreme CBR: Every frame of same size = Extremely restrict budget for I and P frames
- Floating
  - Slightly moving regions can resemble previous frames a lot, until encoder thinks that there is too much difference
    - Instead of slow motion, there will be large jumps
    - Skip mode is PSNR wise a very cheap encoder decision
      - No motion information, guess MV from neighbour
      - No residual to correct anything
      - = Lowest Rate possible
  - An object passes close by a static textured background
    - Background is moved because of the blocks covering object + background

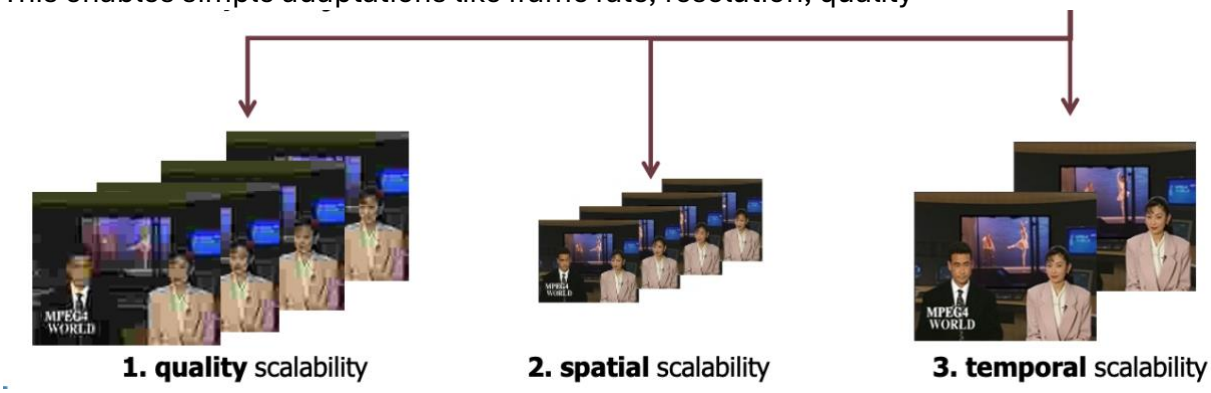
Chapter 10c
Video Coding Extensions

11.9 Congestion management and scalable video coding

Scalable video coding – 3 dimensions

- Quality scalability
- Spatial scalability
- Temporal scalability

This enables simple adaptations like frame rate, resolution, quality



Adaptability – DASH (see H10b)

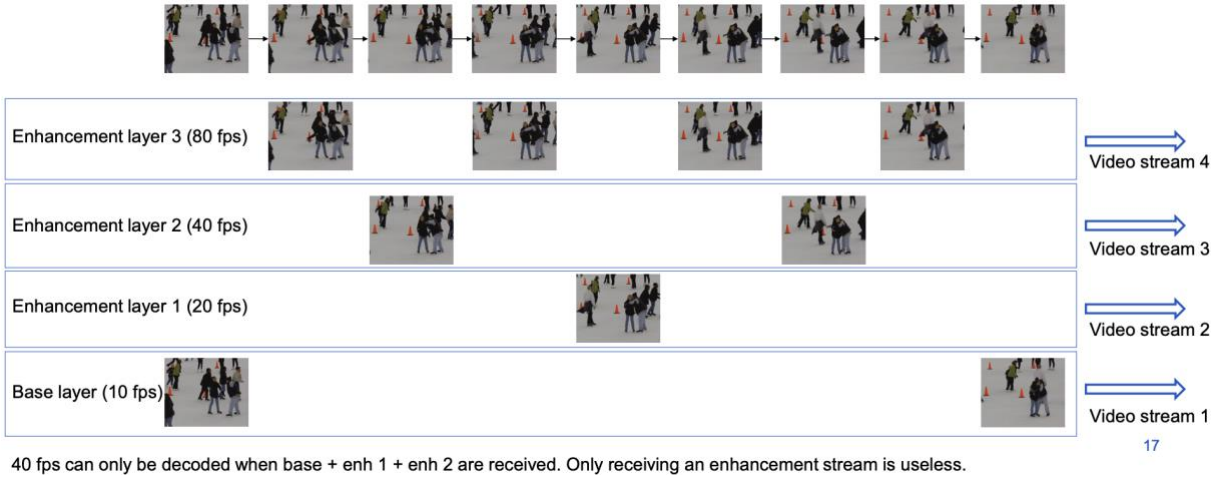
- Adaptable to network conditions
- Diverse end user devices: support varying screen sizes, resolutions

More efficiency

- Save storage = no need to store all possible scales
- Save transmission bitrate

MRF-ME (Markov Random Field) – Hierarchical B prediction

- This enables temporal scalability
- Choose how many layers you will decode to vary the FPS



40 fps can only be decoded when base + enh 1 + enh 2 are received. Only receiving an enhancement stream is useless.

Spatial scalability

- Same concept as for temporal scalability
- Enhancement layer 1 (Full HD) can only be decoded when base layer (SD) + enh1 are received

Multi-loop coding

- Original + down sampled video stream is encoded.
- Down sampled encoded images are upscaled to original size and used in reference buffer of original videostream which are further enhanced
- Same for decoding: decoded downsampled video is used for decoding of HD video

Single loop vs multi loop

H.264/SVC	HEVC, MPEG-2, MPEG-4 Visual
Single loop decoding	**Multi loop decoding**
Complex implementation	Easy implementation
Low processing at decoder	High processing at decoder

- Multi loop
  - Every layer completely decoded
  - Decoded lower res is used to predict the lower frequency components of the higher res picture
  - Reconstructed picture of all levels are stored for each time instant
- Single loop
  - Only target res is fully decoded = ONE motion compensation loop
  - MV, mode and residual data are propagated from lower layers to target
  - A little worse than multi-loop
- Images important! Slides 28-32

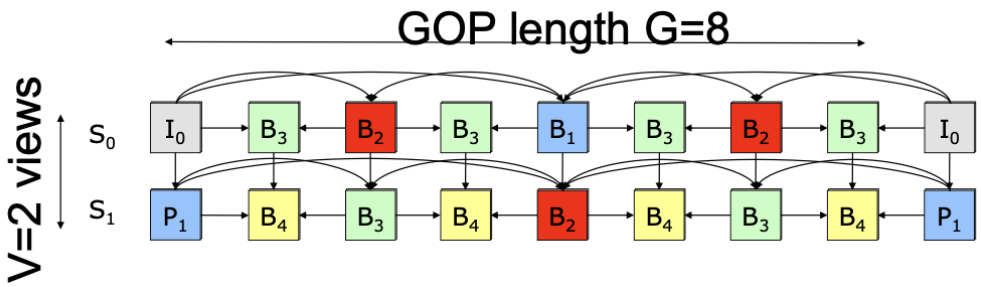
Multiview Video Coding (MVC)

- 3D video – stereoscopic
  - Create illusion of depth
  - Provide eyes with two different images
  - How to efficiently compress both views?
  - Artistic & Technical issues creating 3D video?

- Free viewpoint video
  - Starting from multiple angles of one scene
  - Generate intermediate viewpoints
  - E.g. for interactive replays during sports games

H.264/AVC – MVC

- Stereo High Profile
  - Compression of stereoscopic video sequences
  - Exploiting inter-view correlation
  - More than 2 layers = Multiview High Profile



Multiple Description Coding (MDC)

- Number of independent encodings

Chapter 11
Delivery Across Networks

11.1 The operating environment

Characteristics of modern networks

- TCP = guaranteed delivery with no bounds on delivery time
- UDP = unreliable, with lower overhead: for low delay

Transmission types

- Downloads & streaming
- Interactive communication

Operating constraints

- Limited & Variable bandwidth
- Variable channel quality = transmission errors
- Congestion = packet loss & variable delays

Rate-distortion at encoder is irrelevant for end user

- What matters is quality at the decoder
- Bit streams need to be error resilient, next to RD efficient

Error characteristics

- Random bit errors, often corrected
- Erasure errors (burstly)
    - Can lead to packet loss or retransmission
- Encoding mode greatly influences impact of errors
    - Within a transform block
    - Error propagation due to predictive coding

Challenges

- Error-robust streams require more redundancies
    - Which is what we previously wanted to remove!!
- General approach
    - Understand causes of errors and their effect
    - Use video coding tools to make the encoded stream more robust = application layer(!!)
    - Optimally conceal errors in reconstructed video

11.2 The effects of loss

Synchronization failure

- Bit errors impact entropy decoding (VLC), leading to
  - Wrongly decoded symbols
  - Synchronization loss

Example

- error at position 6
  - symbol sync lost (block sync lost)

Encoded message	B	A	D	E	C	B	A	–
Received bitstream	10	0	1100	1111	110	10	0	
Parsed bitstream	10	0	110	0	1111	110	10	0
Decoded message	B	A	C	A	E	C	B	A

- error at position 10
  - symbol sync regained (block sync will be maintained)

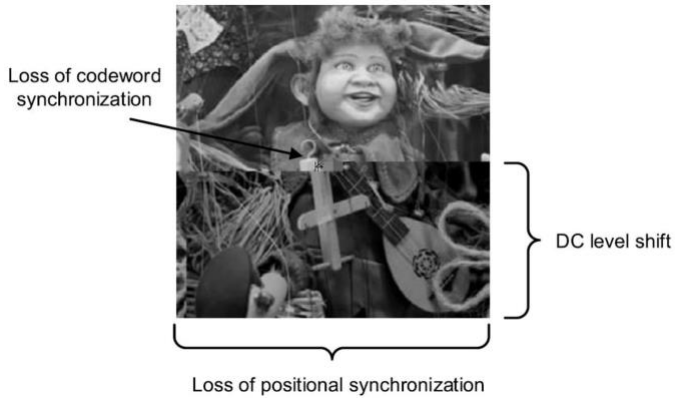
Encoded message	B	A	D	E	C	B	A
Received bitstream	10	0	1110	1101	110	10	0
Parsed bitstream	10	0	1110	110	1110	10	0
Decoded message	B	A	D	C	D	B	A

GHENT

Spatial Error Propagation

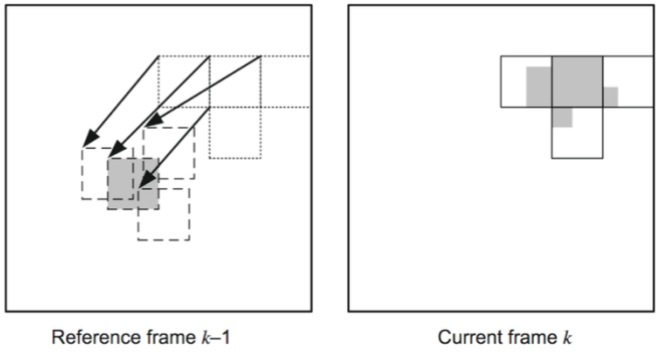
Reasons for spatial propagation

- Loss of VLC sync (incorrect superposition of DCT basis functions)
- Loss of block sync (incorrect EOB symbols)
- DPCM coding (DC coding in JPEG)
- Intra prediction (H.264/AVC intra)



Temporal Error Propagation

- Occurs when corrupted regions are used for prediction
- One block error may influence multiple blocks



11.3 Mitigating the effect of bitstream errors

Video is not the same as data

- If database has error, we won't trust it
- Video has always distortion

Trade-off between rate-distortion and error resilience

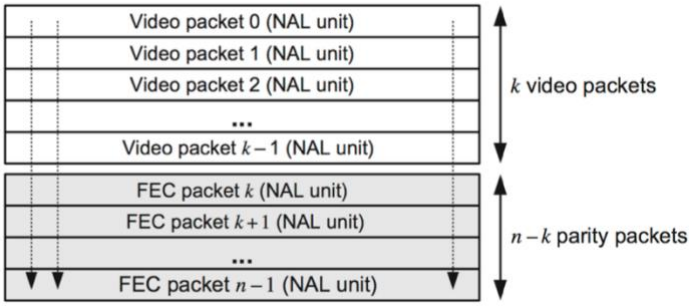
11.4 Transport layer solutions

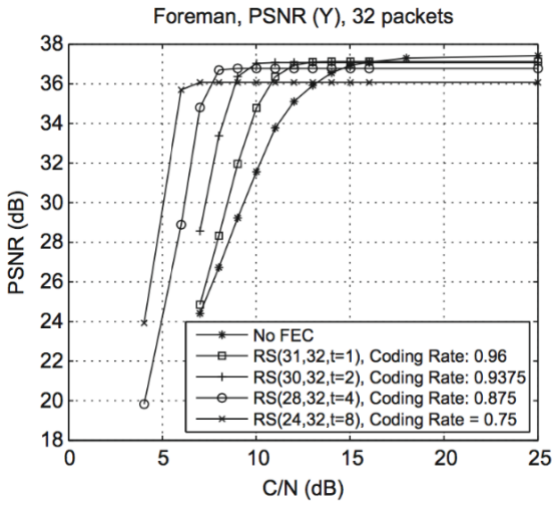
ARQ – Automatic Repeat Request

- Retransmission of packet upon error
- Pro: simple, low overhead
- Cons: delays, requires feedback channel

FEC channel coding (Forward Error Correction)

- Extra parity bits added to the signal
- Unequal error protection in case of data partitioning or layered coding
- FEC assumes worst case scenario regarding errors
    - If channel is worse: codes break
    - If channel is better: bandwidth overhead
- Influence of FEC coding rate:





Packetization strategies

- Longer packet = better for throughput on clean channels
- Shorter packet improves error resilience
    - Less impact
    - Small increase in coding overhead
- Fragmentation of video packets should be avoided

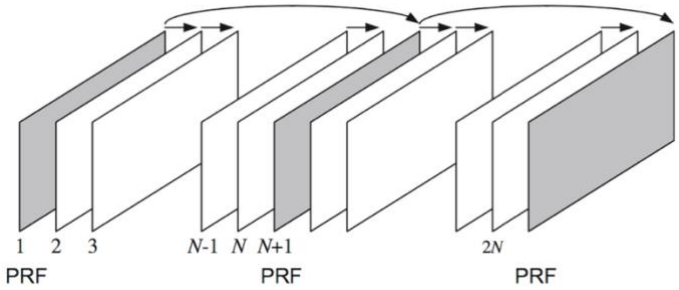
11.5 Application layer solutions

Influence of frame type

- I frames are barriers for temporal error propagation
  - But may introduce spatial errors
- P frames propagate errors from reference frames
  - May introduce spatial errors
- B frames are generally not used as reference
  - Errors are contained within a frame
- Some remarks
  - A P-frame isn't actually defined
  - Any frame can be used as reference

Periodic Reference Frames (PRF)

- More efficient than intra coded frames
- Errors in a PRF might be corrected before it is referenced by the next PRF



Synchronization codewords

- Unique
- Can be inserted at various places (regularly or not)
  - Every N coded macroblocks or every N bits
- Decoding can proceed correctly after such codewords

Reversible VLC

- Bit errors are likely to propagate until an explicit sync symbol
- Reversible VLC codes allow decoding 'from both sides'

Example (a = 0, b = 11, c = 101)

Message	*a*	*b*	*c*	*b*	*a*	*a*	*b*	*c*	*b*	*a*	SYNC
Rx. bitstream	0	11	101	11	**1**	0	**0**1	101	11	0	SYNC
Fwd. decode	a	b	c	b	–	–	X	X	X	X	SYNC
Rev. decode	X	X	X	X	X	X	–	c	b	a	SYNC

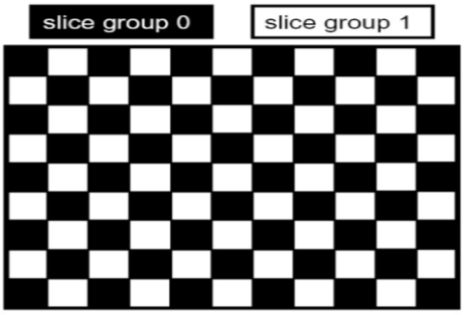
Slice structuring

- Slices contain
  - Slice header
  - Several coded macroblocks

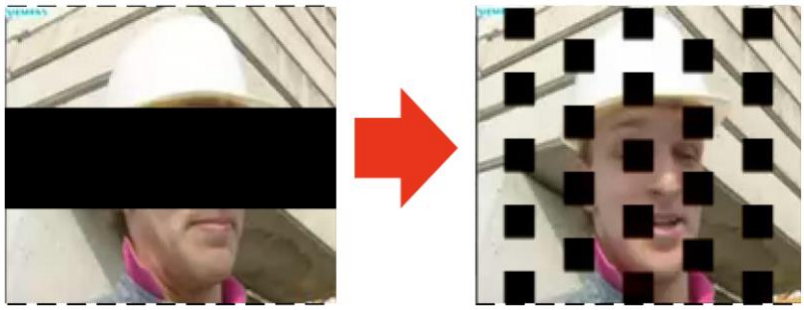
- All forms of prediction are contained within a slice
  - E.g. intra prediction, VLC
  - BUT slices are not 100% self-contained
    - ME restrictions
    - Deblocking filter needs to be disabled on slice edges
- Flexible Macroblock Ordering generalizes the concept of slices to slice groups, enabling arbitrary grouping of coded macroblocks (FMO)
  - Picture => # slice groups => # slices => # macroblocks

Flexible Macroblock Ordering

- Type 1 = dispersed slice groups
- Conceals errors



(b) Type 1



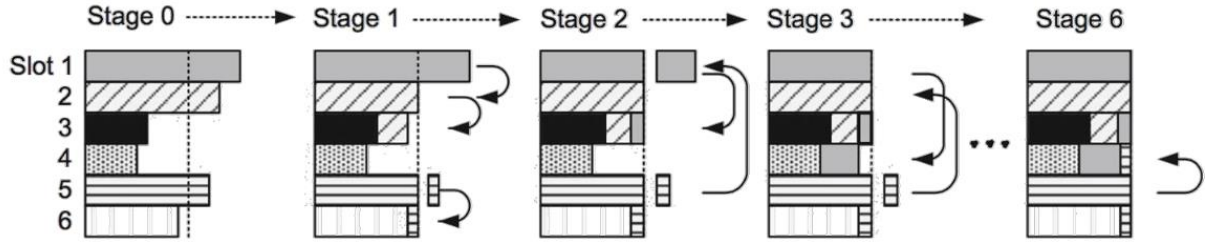
Slice Structuring – continued

- Redundant slices
  - Extra copies of slices inserted into bitstream
- Data partitioning
  - Every NALU (Network Abstraction Layer Unit) is split in three partitions
    - Partition A: slice header, macroblock types, motion vectors, QP, prediction modes
    - Partition B: residual info of intra-coded macroblocks
    - Partition C: residual info of inter-coded macroblocks
  - Can be basis for unequal error protection or different priority cues during transmission

11.7 Inherently robust coding strategies

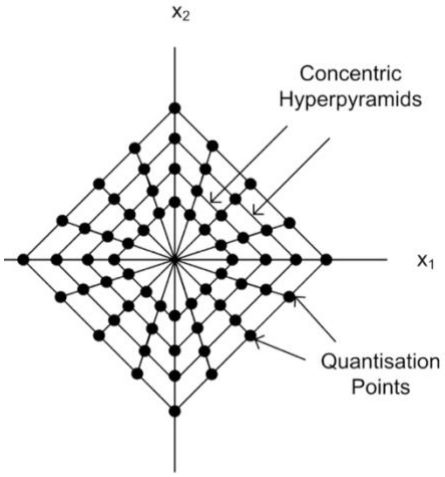
Error-Resilient Entropy Coding (EREC)

- Converts N VLCs to M fixed length blocks of data
  - More resilient to errors
  - Minimal overhead
- General mode of operation
  - N = M
  - Total number of bits (T), block lengths si, and N known to the decoder
  - Bin filling using pseudo random sequence



Pyramid Vector Quantization (PVQ)

- One code vector per encoding region
- For i.i.d. Laplacian random variables
- If they are grouped in L-dimensional vectors, these vectors are localized on a hyperpyramid of dimension L
- Points on hyperpyramid surface have equal probability
  - Uniform distribution of code vectors on that surface
- In practice, dimensions are low
  - Single hyperpyramid would lead to significant distortions
  - Use concentric hyperpyramids



1. Example product code PVQ structure for $L = 2$.

PVQ in practice

- Divide coefficients into categories (e.g. 1-15)
- Group coefficients per category (subbands)
- Every subband coded with different rate and L
- Very similar to wavelet-based coding

1	3	6	8	11	14	14	14
2	5	9	11	11	14	14	14
4	6	9	11	12	14	15	15
7	10	10	12	12	14	15	15
10	10	12	12	12	14	15	15
13	13	13	13	13	15	15	15
13	13	15	15	15	15	15	15
13	13	15	15	15	15	15	15

Type of quantisation

Band 1: Uniform scalar quantiser

Bands 2-5: PVQ with L=16

Bands 6-10: PVQ with L=32

Bands 11-15: PVQ with L=64

11.8 Error concealment

= Post processing at decoder side

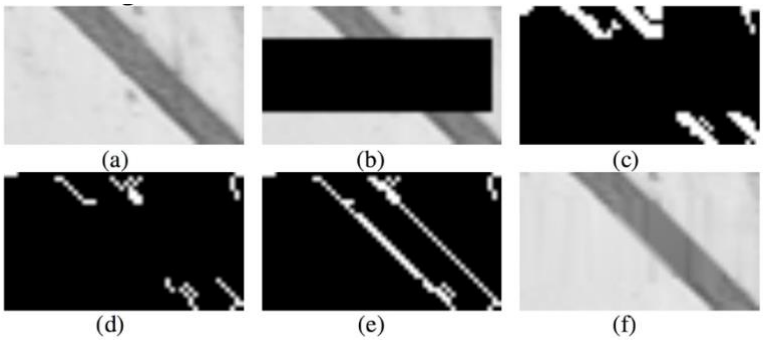
- Goal: estimate the lost info based on previous data
- Can benefit from error resilient properties

First, decoder has to detect that an error has occurred

- Header info
- Error detecting codes
- Erroneous syntax
- Deviating video signal characteristics

Spatial error concealment

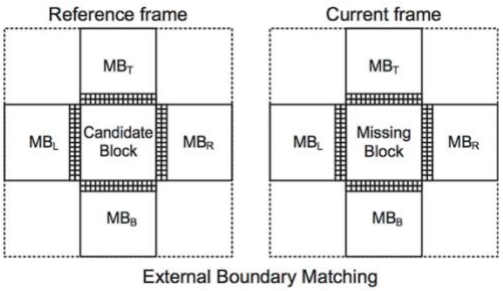
- Exploit special correlation
- Weighted averaging is most used



- Many advanced techniques
  - Based on Sobel filter for edge detection
  - Region-based interpolation

Temporal error concealment

- Naive approach: Temporal copying (of the previous frame)
  - Assumes zero motion
  - Simple method, large errors in moving regions
- More intelligent: motion-compensated temporal replacement
  - Look for a block's best replacement candidate in the reference frame
  - Two stages: estimation of displacement + evaluation
- Estimation of displacement
  - Assuming residual information + MV are lost
  - Several likely candidates
    - MV's from spatially neighboring blocks
    - MV's from temporally neighboring blocks
    - Some combination of the above
  - How to choose a final MV?
    - Cf. matching criterion of traditional MV = use of BDMs like SAD
    - External Boundary matching!



Hybrid methods

- Mode selection based on spatial and temporal activity measures

Chapter 12
Video Coding Standards

12.1 The need for and role of standards

A standard describes an agreed way of doing something
Why use standards as a company?

- Become more competitive by offering products that are accepted globally
- Reduce cost by not reinventing the wheel
- Raise profits by offering products with increased quality, compatibility and safety
- Benefit from the knowledge of leading experts around the world

Why go to standardization meetings?

- Sabotage standard?
- Make early standard ready implementations
- Understand properties of the standard
- Get your patented technology in a standard

What is a successful vide standard?

- Superior performance
- Should allow innovation
- Interoperability (and independence of networks & devices)

Focus on video coding standardization

- Define bitstream format & decoding process
- Standard compliant encoder is no guarantee for quality!!!

Standardization organizations

- Video Encoding Experts Group (VCEG)
- Moving Pictures Experts Group (MPEG)

Recent standards

- Jointly developed
- H.264/AVC, H.265/HEVC, H.266/VVC

Standardization Process

- 4 meetings a year, 10 days each
- Default 36 months
- Lots of stages
- Intellectual property and licensing
    - A loooot of patents, licensors and licensees

12.2 H.120

First digital video coding standard (1984)

- Added motion compensation and background prediction in 1988
- Never commercial success

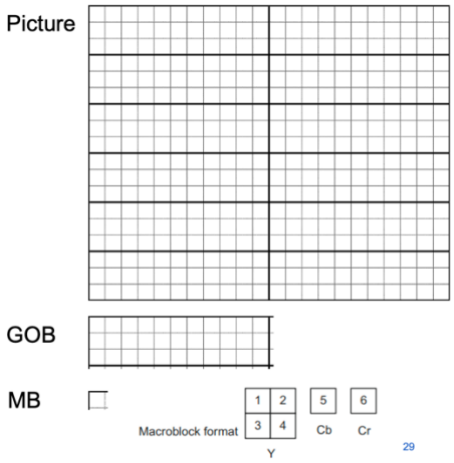
12.3 H.261

First practical coding standard (1989)

- Combining DCT, temporal DPCM (Differential Pulse Code Modulation) and MC
- Block based
- Targeting Integrated Services Digital Network (ISDN) conferencing applications

Fixed structure

- Picture: 176x144 or 352x288 pixels
- Group of Blocks (GOB)
  - Fixed in size
- Macroblock (MB)
  - 16x16
- Block = 8x8



Already very similar to modern codecs

MPEG-1

- Higher rates than H.261
- Added
  - Bi-directional motion prediction
    - But b-frames cannot be used yet as reference
  - Half-pixel motion using large search window
  - slice-structured coding instead of GOB

12.4 MPEG-2

Parts

- Part 1: systems
  - Describes synchronization and multiplexing of audio and video
- Part 2: video
  - Video coding format
- Part 3: audio
  - Audio coding format + multichannel enabled extension + extension of bitrates and sample rates for MPEG-1 audio
- Part 7: Advanced Audio Coding (AAC)

Video

- Developed jointly by ITU-T and ISO/IEC, specifically MPEG and VCEG
- Use for DVD and SD/HD TV
- Newest features
    - Support for interlaced-scan pictures
    - Various forms of scalability

Video profiles and levels

- Profiles
    - Subset of the entire bitstream syntax
    - MPEG-2: 6 Profiles
        - Simple, **Main**, SNR, Spatial, **High 4:2:2**, Multiview
- Levels
    - Range of allowable values for parameters of the bitstream
    - MPEG-2: 4 levels per profile
        - Low: CIF
        - Main: SD TV
        - High1440, High: HD

Profiles and levels

- 1 standard for different use cases and devices

UHD Camera (High4:2:2,High level)

- Need for HD encoding
- Need for >200Mbps encoding
- Intra only must be available
- Capable of 4:4:4

HDTV settop box (Main, High level)

- No need for >HD decoding
- No need for >20Mbps decoding
- E.g. Only progressive content
- Only 4:2:0

- A high profile, high level decoder (HP@HL) can decode
    - HP@HL, HP@ML, SNR@ML, MP@HL, …

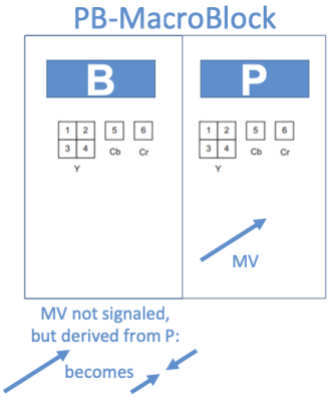
MPEG-2 Video bitstream

- Packetizing the bitstream requires knowledge about the decoding process of the MPEG-2 Visual

12.5 H.263

"The next generation" of video coding (1995)

- Goal of coding at low bitrates
- Initial focus on early mobile radio applications
- Significant impact and use in conferencing, surveillance and internet streaming
- V2 and V3 added many new features



PB-frame

- The B part of the MB, has no MV
- MV derived from P block
- Cheap way to increase frame rate

Motion Vector Prediction

- 4 MV's per MB
- On MB-level and on block-level
- Introduces median MV prediction

Extensions

- H.263+ (1998): 12 new coding modes/tools (Annexes)
  - Error resilience
  - Improved compression efficiency
  - Scalability for resilience and multipoint (multiple RD points)
  - **N**: Reference picture selection (introduction of multiple reference selection)
- H.263++(2000): 3 additional annexes
  - **U**: macroblock- and block-level reference picture selection (!)
  - **V**: data partitioning + reversible VLCs (error resilience)
  - **W**: additional SEI

12.6 MPEG-4

Very ambitious project (toolbox of standards)

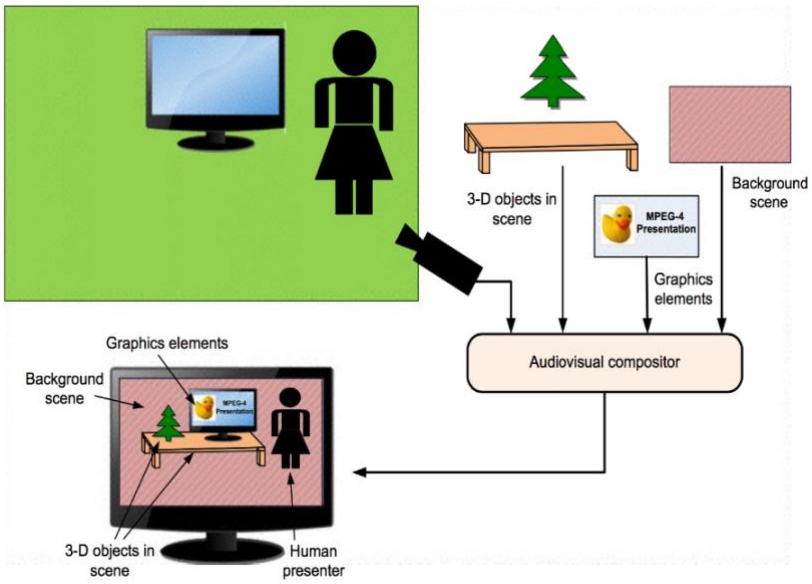
- First version 1999
- Radical new approaches, consists of 31 parts at this point, quite complex and extensive

Most known parts

- Part 2: Visual = SP and ASP
- Part 3: Audio = HE-AAC
- Part 10: H.264/AVC
- Parts 12, 14 & 15: ISO base media file format & MP4/AVC file format

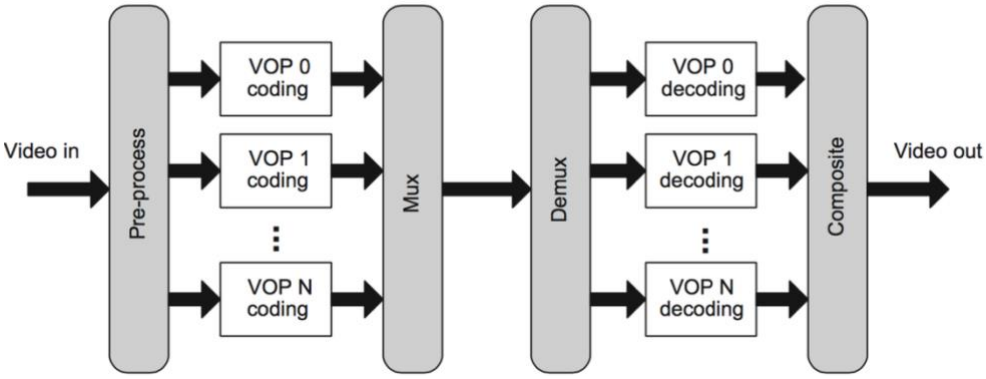
Part 2: Visual

- Global motion compensation
- Quarter-pixel motion compensation
- Unrestricted slices



Example of an MPEG-4 audiovisual scene

MPEG-4 Video Object Plane (VOP) coding:



12.7 H.254/AVC

Most ubiquitous video coding standard to date

- Cf. MPEG-2 for digital video broadcasting
- Mandatory for Blu-ray
- Incorporated in most of the internet streaming sites
- HW support in most mobile devices
- Adopted for HDTV cable, satellite and terrestrial broadcasting

Design Features

- Improved coding efficiency (+50%)
- Network friendly
- Adaptation to delay constraints
- Simple syntax specification

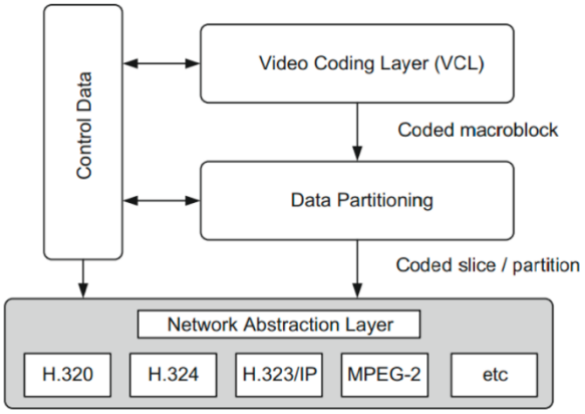
Most of the coding tools already appeared in previous chapters

Profiles

- During the standardization, companies get their own tools, like flexible macroblock ordering, inside the standard
- But these tools are put in a separate profile, which nobody licenses

Network Abstraction Layer (NAL)

- NAL divides these pieces of information in different packets
- SPS and PPS contain resolution, activated tools, …
  - When lost, video undecodable!
- When performing UPD-based transmission, SPS and PPS can be sent separately over TCP



Video parameter set

- Info about how video is constructed
  - Temporal scalability layers
  - Other scalability layers

Sequence parameter set

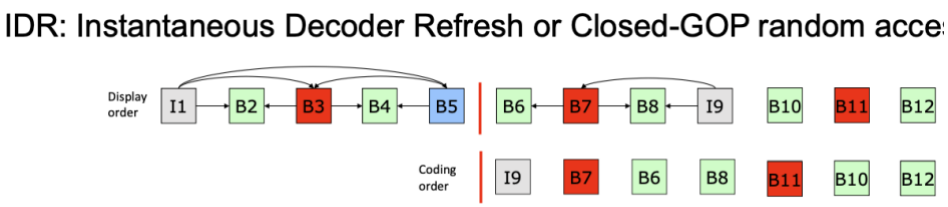
- Info about the entire sequence

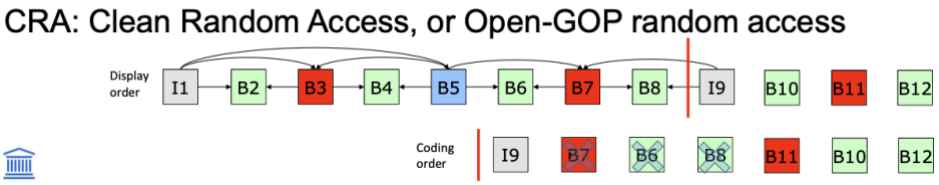
Picture parameter set

- Info about the individual pictures

Random access pictures (RAPs)

- IDR: Instantaneous Decoder Refresh or Closed-GOP random access
  - A frame belonging to a closed GOP can only refer to frames within its own GOP
  - When the decoder encounters an IDR frame, it can flush its picture buffer (Decoded Picture Buffer or DPB)
- CRA: Clean Random Access or Open-GOP random access
  - Allows frames from an open GOP to refer to frames from another GOP
  - More compression efficiency

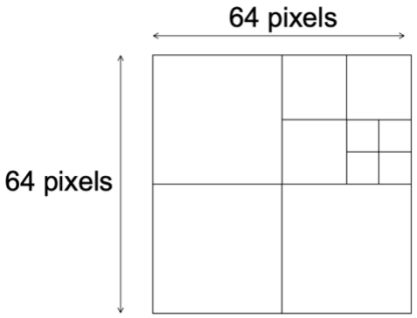




12.8 HEVC (H.265)

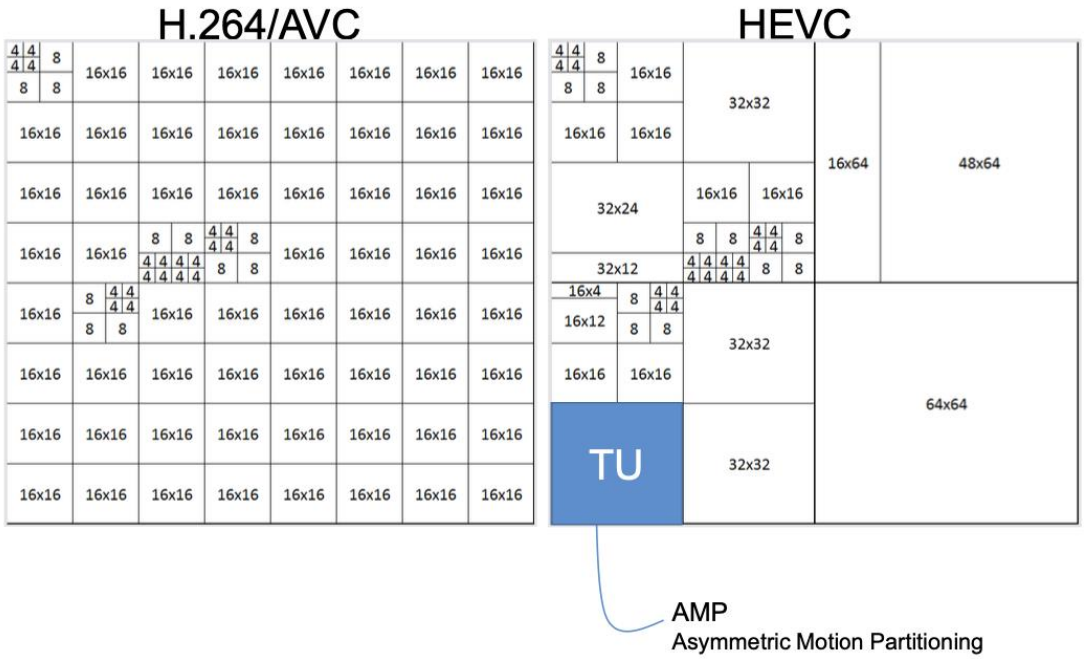
High Efficiency Video Coding (HEVC)

- Successor to H.265/AVC
- V1 since 2013
- Significantly better than H.254/AVC
    - Same quality for half the bit rate!!



Coding Tree Unit (CTU) partitioning

- Partitions from 64x64 to 8x8 pixels
- Also
    - coding unit (CU), 32 x 32 left of CTU
    - Prediction Unit (PU), 16 x 64 Block within CTU
    - Transform Unit (TU)
        - Asymmetric Motion Partitioning (AMP)



Slices

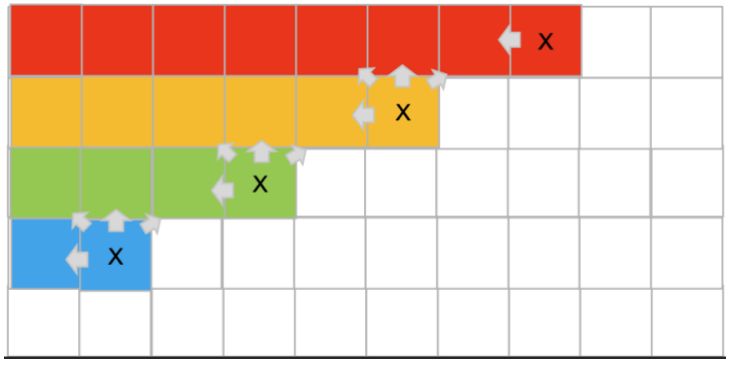
- A succession of blocks (CTUs) in encoding order, which are independently coded as one unit
    - You are not allowed to use any information from outside the slice, except from reference pictures

Slices enable

- Parallel and independent encoding. Parallel decoding cannot be assumed!
- Resynchronization and error robustness
- Constant NALU size
- Low-latency streaming, because processing of small units of data independently, faster encoding and decoding

Wavefront Parallel Processing (WPP)

- The internal state of the context variables is not carried over to the start of a CTU row from the right-most CTU in the previous row, but rather from the second CTU in the previous row
    - Better compression efficiency compared to tiles and slices
- Multiple CTU rows to be processed in parallel for decoding
- Lossless transcoding possible between WPP and non-WPP streams



Intra prediction evolution

- 9 directions in AVC, to 35 in HEVC

Motion vector prediction

- AVC = median of neighboring vectors
- HEVC = Advanced Motion Vector Prediction (AMVP)
    - Derivation of at most 2 probable candidates
        - Left predictor
        - Upper predictor
        - Temporal predictor
    - Merge: no MV refinement

Large-size transforms

- AVC = 8x8, HEVC = 32x32

Loop Filters

- Deblocking filter (DF)
    - Improve visual quality + prediction performance
    - Smoothing sharp edges between transform units
    - ~Low Pass filter
- Sample Adaptive Offset (SAO)
    - Classify reconstructed pixels into different categories
    - Reduce distortion (restore edges) by adding an offset for each category of pixels
    - Applied after the deblocking filter
    - ~High Pass filter

Summary

H.261	MPEG-1	MPEG-2 H.262	H.263	MPEG-4	H.264/AVC	HEVC (H.265)
CIF 352x288 4:2:0	≤ 4kx4k @30fps	Theoretically 16kx16k 4:2:2, 4:4:4	*≤16CIF*	**Legend:** Left to right: features added by standard *Italic: one-time features*		
-I, P-pictures -No intra pred	Unref. B-picture (b)		*PB-picture*	*Object coding*	-Ref B-picture (B) -9 intra pred.	35 intra pred.
GOB=fixed size	*Unrestricted slice*	<= 1-row slice	GOB=[1,2,4]-row slice	Unrestricted slice		Slices, tiles and wavefront
8x8 DCT					4x4, 8x8 Integer DCT	4x4 - 32x32 integer DCT, DST
±15 search window 1MV/MB	±512 search window ½ pixel MC		-unrestricted search window (outside) -median MV predict 8x8 – 16x16	*Global MC* ¼ pixel MC	Multiple Ref 4x4 – 16x16	 2 candidate MV prediction 4x4 – 64x64
VLC			Arithm. coding		Context adapt. binary arithm coding CABAC	
		Multi-loop scalability (Temp., SNR, Spat.) Profile, Level	Deblocking filter		NAL *Single-loop scalability*	Multi-loop scalability SAO: Sample Adaptive Offset

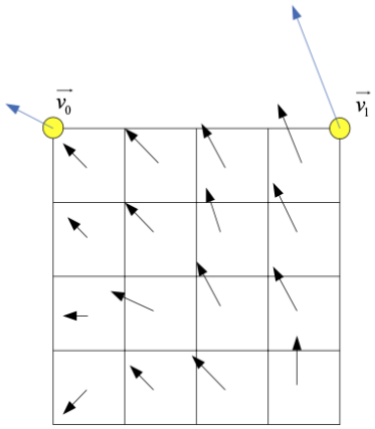
12.9 VVC (H.266)

Versatile Video Coding (VVC)

- Even better than HEVC
- Beyond the standard and high definitions
  - Flexible and large block structures (CTU 128x128) see image
    - Subblocks can be split in two or in three parts
  - Affine motion compensated prediction
- Computer-generated or screen content
  - Intra-picture Block Copy (IBC)
- Ultralow-delay streaming
  - GDR: Gradual Decoder Refresh (additional to IDR and CRA)
  - Reference picture resampling (RPR)
- 360 video
  - Wrap around motion compensation

Affine motion compensated prediction

- Zoom, rotation, perspective motions and other irregular motions
    - Affine motion using 2 or 3 control point MV's

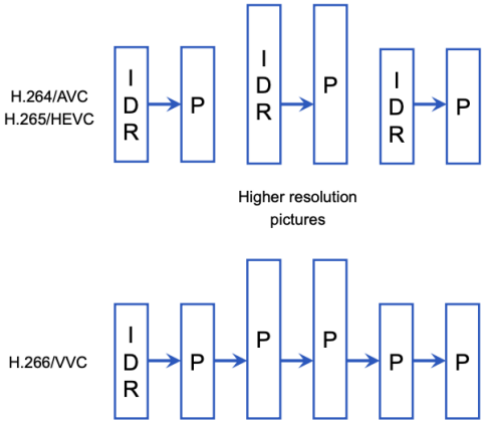


Gradual Decoding Refresh = Intra refresh

- Blocks are refreshed (intra coded) in a horizontally scrolling column – the 'refresh wave'
- + low-latency streaming: more constant frame sizes than with standard IDR-frames
- + increase resilience of the video stream to packet loss
    - Because of I blocks
- - Reduce compression efficiency, hence only use it when needed
    - Loss comes primarily from the fact that blocks on the 'new' (left) side of the refresh wave can't refer to data on the 'old' (right) side

Reference Picture Resampling

- Changing resolution on a frame-by-frame basis
- No need for IDR pictures anymore when switching resolution
- Ultralow-delay streaming: adapt resolution to available bandwidth without bitrate spike of IDR picture



Other standards

- Mainly Google (VP9, VP10)
- AV1
    - In between VVC and HEVC
        - Intra prediction directions
        - CTU = 128 x 128
        - Inter prediction
            - 1/8 pixel MC = HEVC
            - Affine = VVC

Chapter 13a
The Future

13.2 New formats and extended video parameter spaces

Spatial Resolution

- Number of pixels is not that important
- Key parameter = angle between pixels, as seen from an observer
    - Resolution, screen size and viewing distance are related
- There is no such thing as HD quality
    - More pixels can bring more detail
    - Compression system must be allowed to retain those details
    - Very easy to turn HD into SD through compression

UHDTV – Colorspaces

- SD uses ITU-R Rec. 601
- HD uses ITU-R Rec. 709
- UHD uses ITU-R Rec. 2020
- If 601 is decoded as 709 => Red too orange, Green too dark
- If 709 is decoded as 601 => Red too dark, Green too yellowish

Table 13.1 Parameter set for UHDTV/ITU-R Rec.2020.

Parameter	Values	HDTV (Rec. 709)
Picture aspect ratio	16 × 9	16:9
Pixel count ($H \times V$)	7680 × 4320, 3840 × 2160	1920x1080
Sampling lattice	Orthogonal	orthogonal
Pixel aspect ratio	1:1 (square pixels)	1:1 (square)
Frame frequency (Hz)	120, 60, 60/1.001, 50, 30, 30/1.001, 25, 24, 24/1.001	60…24
Color bit depth	10 or 12 bits per component	8 or 10 bit per component
Scan mode	Progressive	progressive & interlaced
Viewing distance	0.75H	3H
Viewing angle	100°	ca. 33°

Compression Performance

- Increased spatial resolution is useless unless the compression system can preserve the improvements

Temporal Resolution

- There is a growing mismatch between frame rate and spatial resolution
    - May cause peripheral flicker on large screens
    - Fix: temporal upsampling inside TVs (300-600fps) for smooth motion
- Portrayal of motion is trade-off between
    - Long shutters (motion blur)
    - Short shutters (jerky motion and temporal aliasing)

Static and Dynamic resolution

- Effective spatial resolution of a format in the presence of motion

  Example 13.1
  - object moves left to right in 4s at 30 Hz (i.e. in 120 frames)
  - SD (720xH_1), 20° viewing angle
    - statically, there are 36px per degree
    - object 'travels' 6px per frame (motion blur) = 720/4/30
    - → dynamic resolution of 36/6 = 6 px per degree
  - UHDTV (7680xH_2), 100° viewing angle
    - statically, there are 76.8 px per degree
    - object 'travels' 64 px per frame (motion blur) = 7680/4/30
    - → dynamic resolution of 76.8/64 = 1.2 px per degree

If frame rate doubles, bitrate doubles?

- + Temporal correlation increases
  - Smaller MV magnitude
  - More correlated MV's
- + Motion better fits translational motion model assumed in ME
  - Reduced residual energy
- - More high-frequency spatial detail (less motion blur)
  - Harder to code
- In general: frame rate increase of x% = bit rate increase of x/2%

PSNR vs MOS vs FPS

- Higher FPS grants overall lower PSNR for equal bitrate
- But gives a higher MOS
  - Bigger difference for higher motion sequences

Reality = High Dynamic Range

Luminance => There is no limit on the intensity of light

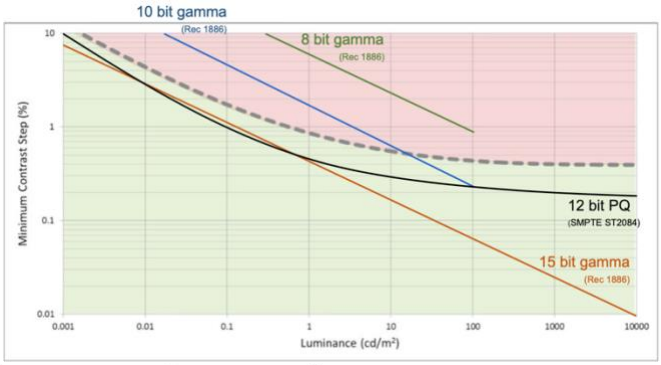
- Problem: There is a limit on how much light your display can produce
- **nit** is a non-SI name also used for cd/m2 (cd = candela)

Contrast ratio

- Human eye without adaptation
  - 100.000 : 1
    - The contrast between dimmest light a human can see and the brightest
- Video signal
  - 8, 10 or 12 bit = 4096:1
- Display
  - Max 11000:1
  - OLED = inf:1 because it uses black pixels for black color
- How to map limited bitdepth to high dynamic resolution of display?

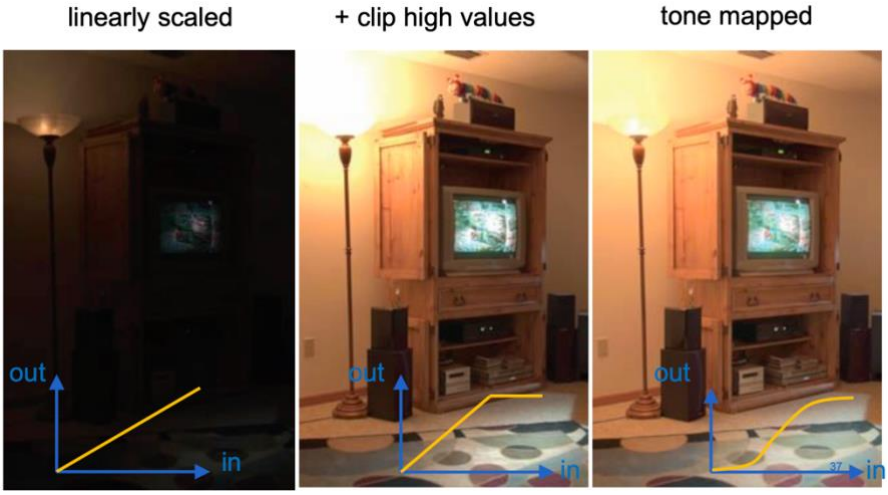
Limited signal bitdepth => EOTF

- Electro-Optical transfer function
- Barten ramp (see image)



Limited screen luminance => tone mapping

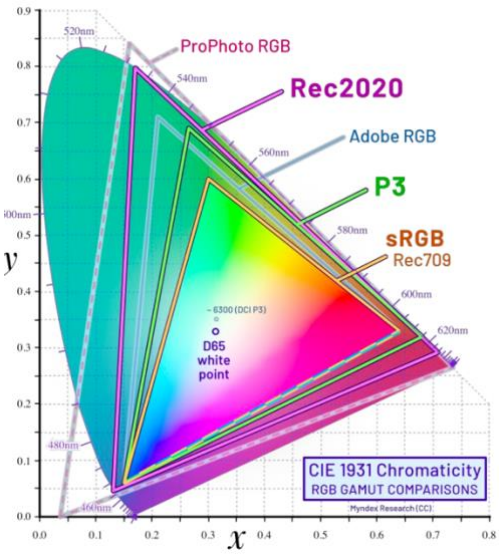
- Conversion from high dynamic range to lower dynamic range



HDR effect on phone is not HDR, it's tone mapping

Wide Color Gamut (WCG)

- Screens are restricted to their own gamut
- Rec.2020 uses real colors, ProPhotoRGB does not



Chapter 13b
The future

13.1 Immersive experiences

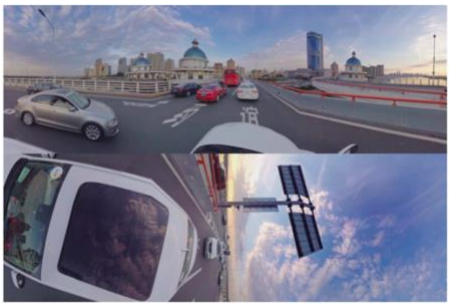
Plenoptic function(?)

360 degree video
- 6-DOF using camera content
- L(x,y,z,theta,phi) => L(theta,phi)
    - Make it spherical

Projection mapping for 360 video
- Depending on the projection format used to represent the spherical video
    - Different areas of the sphere are sampled at different densities

Equi-rectangular projection mapping (ERP)
- Most widely used for representing 360 video on a 2D plane
- Oversamples the sphere at the poles, resulting in stretched top and bottom areas

Equal-area projection format (EAP)
- Solves oversampling at poles using arcsin function



Cubemap projection format (CMP)
- Split video in two: top and bottom
    - Uses 6 sides of a cube, stitched together

Segmented sphere projection format (SSP)
- Segments the sphere in 3 segments
    - North pole
    - Equator
    - South pole

Stereoscopic 360 video
- Single camera decrease parallax effect
    - = Bad feeling of movement (close and far)
    - Need multiple cameras to fix this!
- Multiview video coding
- Example: 6x Fish eye camera in a circle, take 60 degrees from each camera pair and stitch them

Problem 3 – stereoscopy (VR brillen enal)

- The natural reflex to change focus with depth should be suppressed
  - = Vergence Accomodation Conflict (VAC)
- Solution: Light Fields
  - At every position on the display, different light needs to be projected in different directions
  - E.g. Less divergent light for objects that are further (see slide 45)
  - Practically: placing micro lenses in front of high-res display

Light field capture devices (so you can display it later)

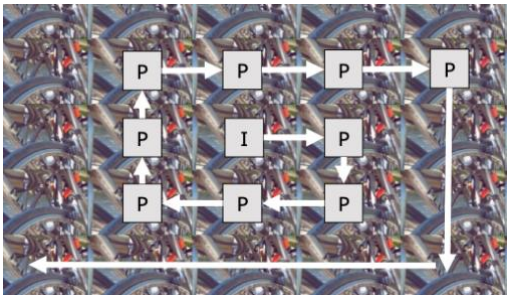
- Dense light field: micro lenses in front of high-res sensors
- Sparse light field: Taking images with multiple cameras

Light Field rendering

- Rendering a 2D view from a 4D light field
- This is done after capturing

1. Walking around
   a. Requires calculating relevant light rays = light field rendering
2. Choose aperture – depth of field
   a. Larger apertures can be calculated by combining the images from more cameras = make everything as sharp as possible
3. Choose focus distance

- Interpolate unmeasured light rays



Light Field Coding using traditional codecs

- Temporal prediction applied to light field images
- Multiview video compression applied

Coding by modelling a continuous representation

- Neural Radiance Fields (NeRF)
  - Deep neural network
  - Input = 5D coordinate (location (x,y,z) + view (theta, phi))
  - Output = volume density + view dependent emitted radiance
- 3D Gaussian Splatting
  - Like NeRF but also with color + transparency
  - Uses Probability to guess colors using spherical harmonics

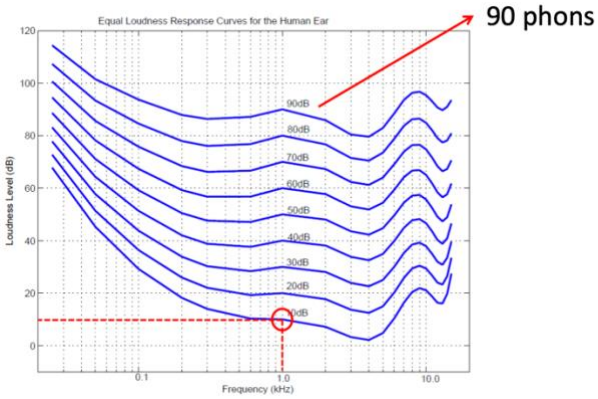
Chapter 14
MPEG audio compression

Remember Differential coding of audio samples of a wave form

- Limited R-D performance
- Compared to video coding, we observe that these rudimentary compression techniques
    - Fully operate in time domain (no transform)
    - Don't have optimizations w.r.t. Human Auditory System

Psychoacoustics

- Range of human hearing is 20Hz to 20kHz
- Freq of voice is 500Hz to 4kH
- Dynamic range is on the order of about 120dB
    - = Ratio of loudest sound to quietest sound a human can hear
- Magnitude of sound [dB] = SNR = 20*log (Vsignal/Vnoise)
- Image = Threshold of human hearing

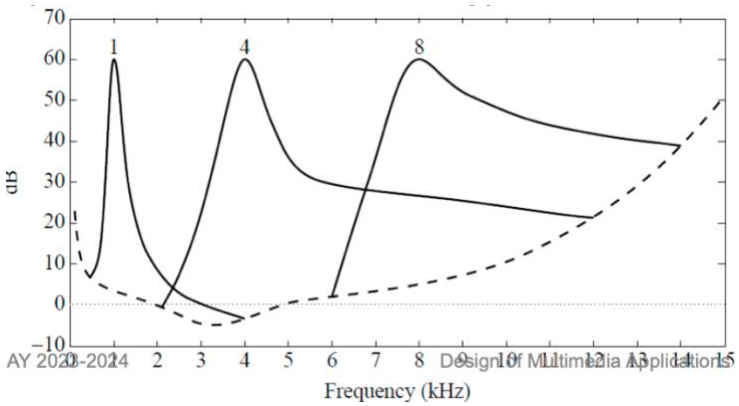


Fletcher-Munson Curves

- Ear is more sensitive in the range of 2-5 kHz
- Equal perceived loudness curves
    - Equal to tone at 1kHz

Frequency masking

- Lossy audio compression like MPEG Audio encoding, removes some sounds which are masked anyway
- General situation
    - A given tone can effectively mask neighboring freqs
        - Making the others inaudible
        - Not symmetric: masks higher freqs more efficient than low freqs
    - The greater the power in the masking tone, the wider its influence
- Example: Masking curve for 1, 4, 8kHz masking tone
- Higher masking tone = broader influence



Practical implications for compression

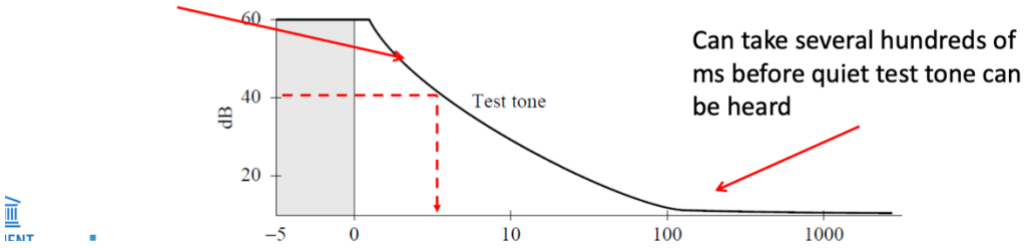
- For frequencies that will be partially masked, only audible part will be used
- Quantization noise is acceptable as long as it is masked by the signal

Critical Bands

- Represents the ear's resolving power for simultaneous tones or partials
- Human auditory system cannot resolve sounds better than within one critical band when other sounds are present
    - Hearing has a limited, frequency-dependent resolution
    - In a complex tone, the critical bandwidth corresponds to the smallest frequency difference between two partials such that each can still be heard separately
- Constant volume sound seems louder if it spans multiple critical bands
- At low freq, critical band is smaller than at high freq
- Implication
    - Ear operates like a set of bandpass filters
- Audio frequency range for hearing can be partitioned into 24 frequency bands

Temporal masking

- Phenomenon: Any loud tone will cause hearing receptors to become saturated and require time to recover
    - Louder test tone = shorter amount of time required before test tone is audible once masking tone is removed
    - The longer the length of masking tone, the longer it takes before test tone can be heard. Curve will be bol instead of hol
        - e.g., 1 kHz masking tone at 60 dB and *1.1 kHz* test tone at *40 dB* → takes several milliseconds before test tone can be heard



Effect of Temporal and Frequency masking depending on both time and closeness of frequency

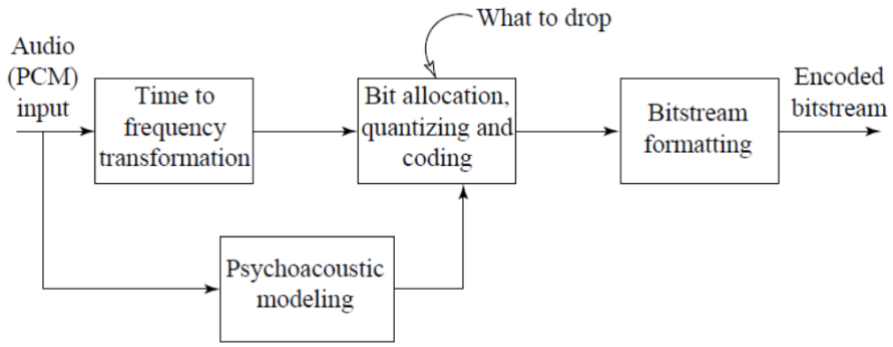
- The closer the freq of test tone to the masking tone and the closer in time to when the masking tone is stopped, the greater the likelihood that a test tone cannot be heard
- Test tones at freq near masking tone are masked the most

MPEG Audio Strategy

- Employs several filters to:
  - Analyze the frequency components of the audio signal using freq transform
  - Decompose the signal into subbands
    - Layer 1&2: “Quadrature-mirror”
    - Layer 3: adds a DCT
- Relies on
  - Quantization
  - Human auditory system is not accurate within a critical band
- Components that are masked by frequency or temporal masking are transmitted with fewer bits

How it works – overview

- Applies filter bank to the input to break it into its frequency components
- In parallel, psychoacoustic model is applied for bit allocation
- The number of allocated bits are used to quantize the information from the filter bank
  - Providing compression
- For each frequency band (32), sound level above the masking level indicates how many bits are to be used so that quantization noise is below the masking level

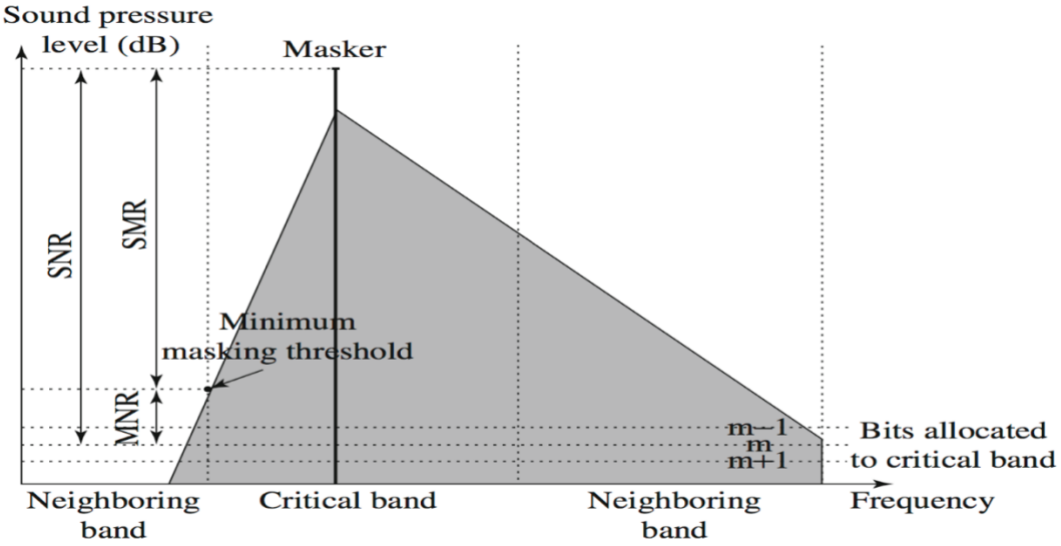


Basic algorithm

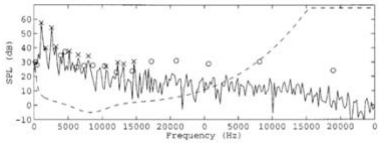
- Divide PCM input into 32 frequency subbands
- Output is 32 frequency coefficients
- In the Layer 1 encoder, the sets of 32 PCM values are first assembled into a set of 12 groups of 32 samples
  - This gives time lag in the coder
- Output of psychoacoustic model is encoder-dependent
  - Set of signal-to-mask ratios (SMR) that flag freqs with amplitude below masking level

Bit allocation

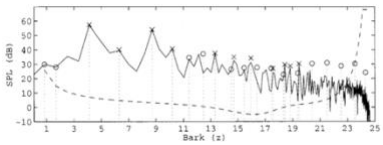
1. For each subband, calculate SMR = 20 * log (signal / min_mask_th)
    - This determines quantization for signals above threshold
2. Calculate signal-to-quantization-noise ratio (S(Q)NR)
    - Through lookup table
3. Calculate mask-to-quantization-noise ratio: MNR = S(Q)NR – SMR
4. Iterate until no bits left to allocate
    a. Allocate bits to the subband with the lowest MNR
    b. Look up new estimate of SNR and calculate MNR



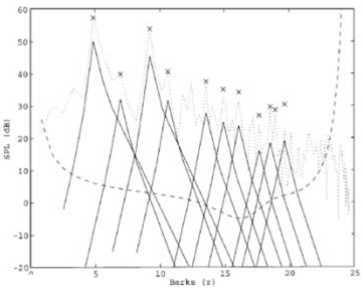
Example: MPEG-1 Psychoacoustic Model 1



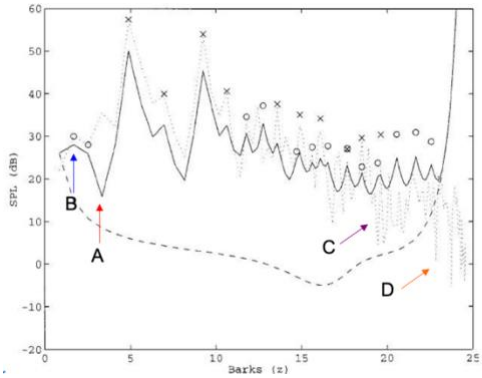
1. Spectral analysis and SPL normalization



2. Identifications of **tonal** maskers
   and calculation of individual masking thresholds
3. Same but for **noise** maskers (circles)



4. Calculation of global masking thresholds
5. Bit allocation



A - Some portions of the input spectrum require SNR's > 20 dB

B - Other portions require less than 3 dB SNR

C - Some high frequency portions are masked by the signal itself

D - Very high frequency portions fall below the absolute threshold of hearing.

MPEG-1 Audio offers three compatible layers

- Each succeeding layer
    - able to understand the lower layers
    - offering more complexity in the psychoacoustic model + better compression
    - with increased compression, comes extra delay
- Objective = tradeoff between quality and bit rate
- Layer 3 = MP3

MPEG-2 AAC (Advanced Audio Coding)

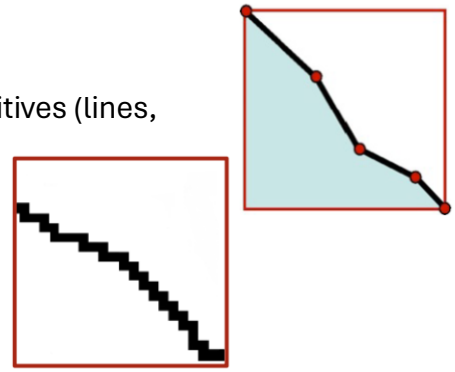
- Standard for DVDs
- Aimed at sound reproduction for theaters (5 channels for stereo effect)

Chapter 15
3D Mesh Coding & Compression

3D Graphics Rendering
- Two main components
  - Textures: information for visual appearance
  - 3D ‘models’: representation of ‘surfaces’ (like triangles)

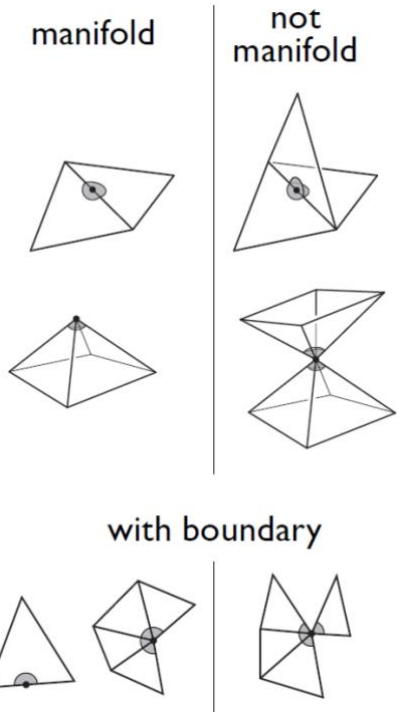
Surface can be modeled
- Parametrically
- Implicitly
- Finding a fitting function for a complex 3D model is impossible
  - Solution = SAMPLING

Modeling by sampling
- Piecewise parametric
  - 2D = combination of known 2D primitives (lines, curves, polygons)
  - 3D similar using 3D primitives
- Piecewise implicit
  - 2D = pixels (picture elements)
  - 3D = voxels (volumetric elements)



Scanning a 3D model: Discretization
- Triangles!
  - Defined by three vertices
  - Lives in the plane containing those vertices
  - Vector normal to plane is the triangle’s normal

Triangle Meshes
- A bunch of triangles in 3D space that are connected to form a surface
- Mesh = piecewise planar surface
- Creases between triangles are artifacts



Manifolds (‘Surfaces’)
- Topological space that locally resembles Euclidian space near each point
- 2-manifold: each point of a 2-dimensional manifold has a neighborhood that is homeomorphic to the Euclidian space of dimension 2
- Properties
  - Each edge must have exactly 2 triangles
  - Each vertex must have one loop of triangles

Euler identity, valid for all 2-manifold orientable meshes

- v – e + f = 2 * (s – g) – b = X
  - v = number of vertices
  - e = number of edges
  - f = number of faces
  - s = number of connected components
  - g = the genus = number of handles
  - b = number of boundary loops
  - X = Euler characteristic often = 2 (closed object, genus 0)

Mesh is represented as

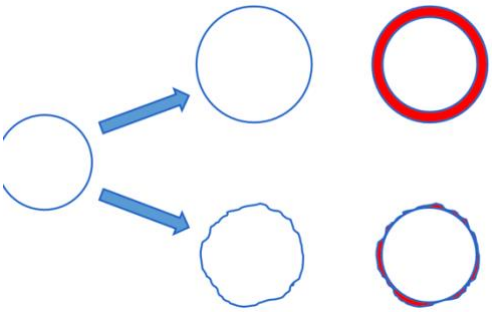
- Connectivity = how are vertices connected
- Geometry = what samples are used
- Together they determine the surface
- What to encode (see image)

Image	Mesh
	Connectivity
	Geometry
Color	(Possibly) Attributes

Vertices (i.e., geometry)	Triangles (i.e., connectivity)
v_1: x_1,y_1,z_1	v_1, v_2, v_3
v_2: x_2,y_2,z_2	v_3, v_2, v_4
v_3: x_3,y_3,z_3	…
v_4: x_4,y_4,z_4	…
…	…

How to handle complex models

- Generate several “Levels Of Detail” (LOD)
- Only load appropriate level
- Similar to having multiple resolutions of a video



Measuring mesh errors

- Largest point to surface distance
- But: Large objective distance does not always imply large subjective distortions
- Evaluation even more critical for interactive use
  - We do not know what parts the user will look at

Geometry size:

$$3v\ 32bit$$

Connectivity size:

$$3f\lceil \log_2 v \rceil bit$$

Geometric Quantization

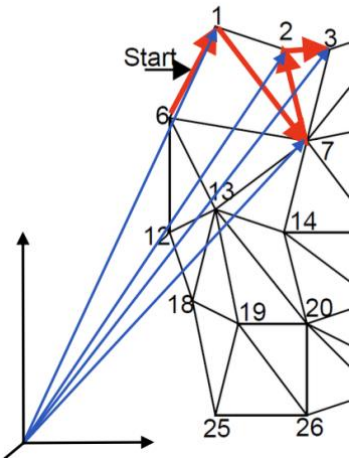
- Without:

Geometry: $96bpv$
Connectivity: $6\lceil \log_2 v \rceil\ bpv$

- With geometric quantization, 16 bit suffices (= /2)

Geometry: $48bpv$
Connectivity: $6\lceil \log_2 v \rceil\ bpv$

Can we do more?



- Group triangles together using fans and strips
- More geometry compression!
  - Difference between subsequent vertices (red) is much smaller than 16 bit (blue)
- Touma&Gotsman
  - Pick initial triangle
  - Mark focus vortex
  - Count for every vertex amount of neighbours
  - Add them
  - Expand with neighbouring vertex of the focues vertex, add to list
    - Go around the focus vertex until done
  - New focus vertex = next vertex
  - If next free edge is already in active list
    - Split active list in two (Keep original focus, new focus on the 'next free edge')
    - Smaller active list is pushed on stack
    - Finish larger active list
    - Come back to smaller active list
  - If (sub)list is compelely done, add amount of vertices of that list (Dummy)