H.264 Post Processing for Flicker Reduction

Dimitry Kletsel, Yair Kuszpet
Supervisors: Yair Moshe, Dr. Avi Levy
In association with Intel-Oplus

18/07/2007

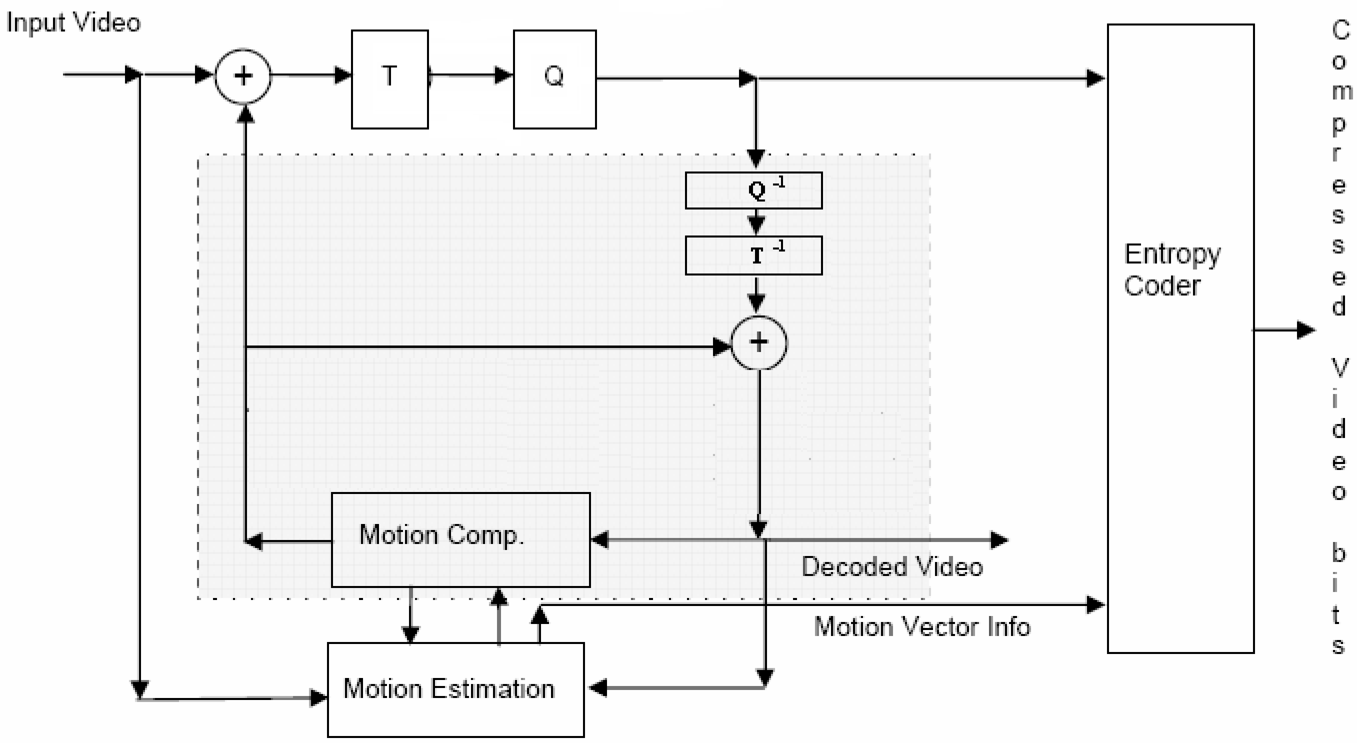
Outline

- Video Coding. MPEG2 and H.264
- Coding Artifacts
- Temporal Artifacts in MPEG2
- The Temporal Flicker Artifact in H.264
- Nonadaptive Approach to Flicker Reduction
- Adaptive Approach to Flicker Reduction
- Results, Conclusion, and Future Work

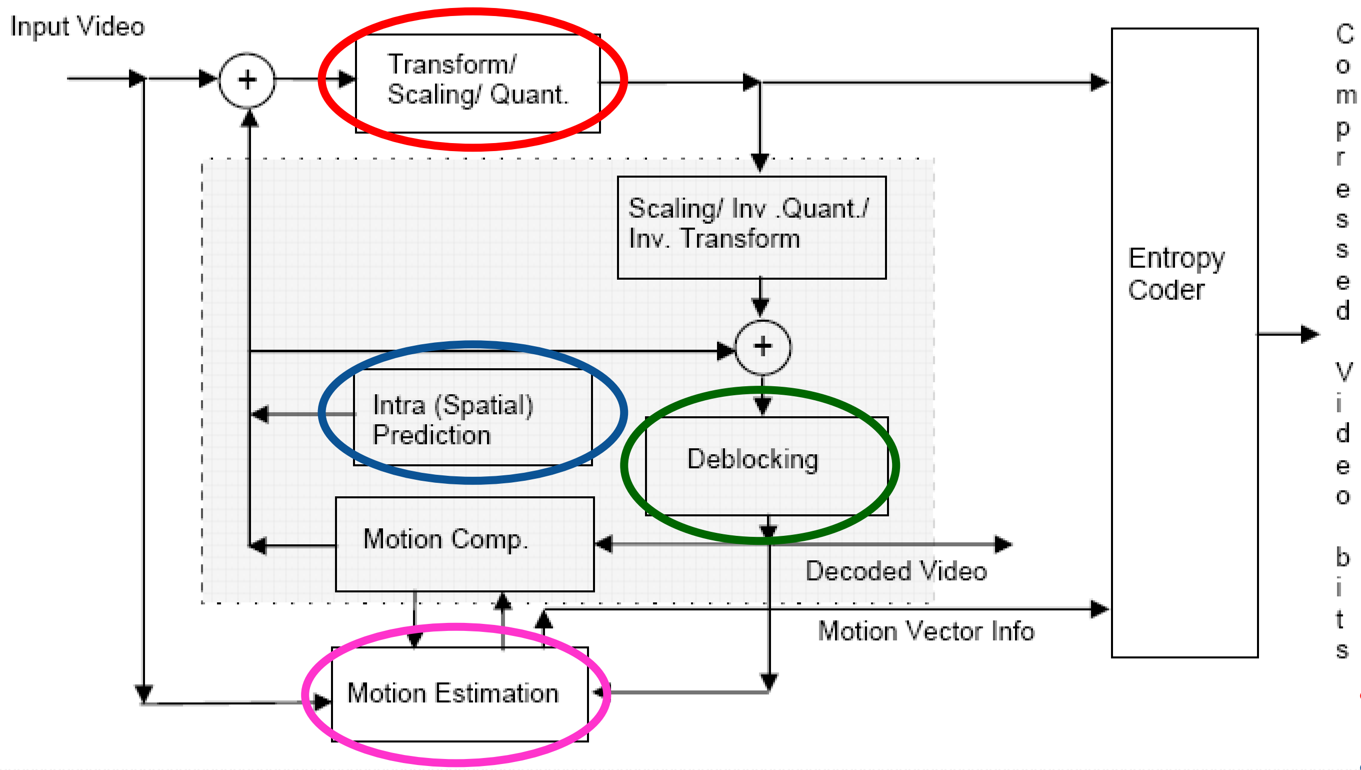
Project Goal

- Characterization of H.264-encoded video artifacts
- Development and examination of suitable Post-Processing techniques

MPEG2 Block Diagram



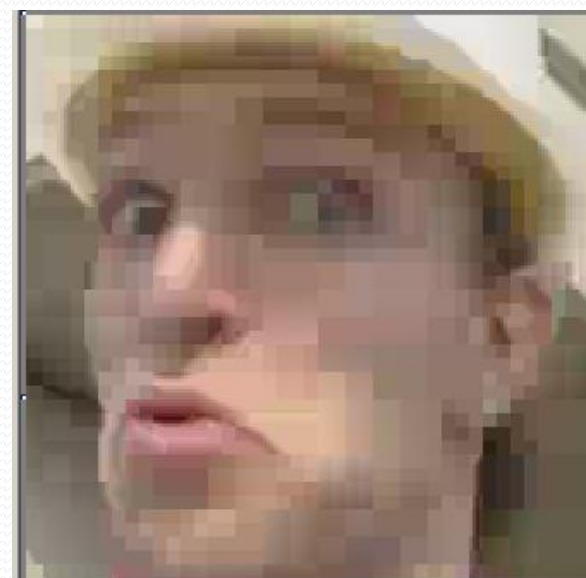
H.264 Block Diagram



- ICT
- Intra prediction
- Deblocking filter
- Motion Estimation

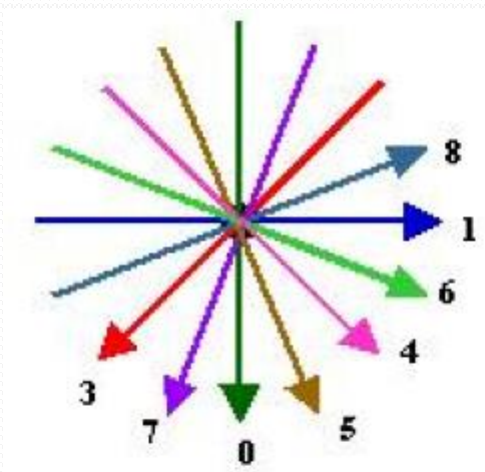
H.264 Deblocking Filter

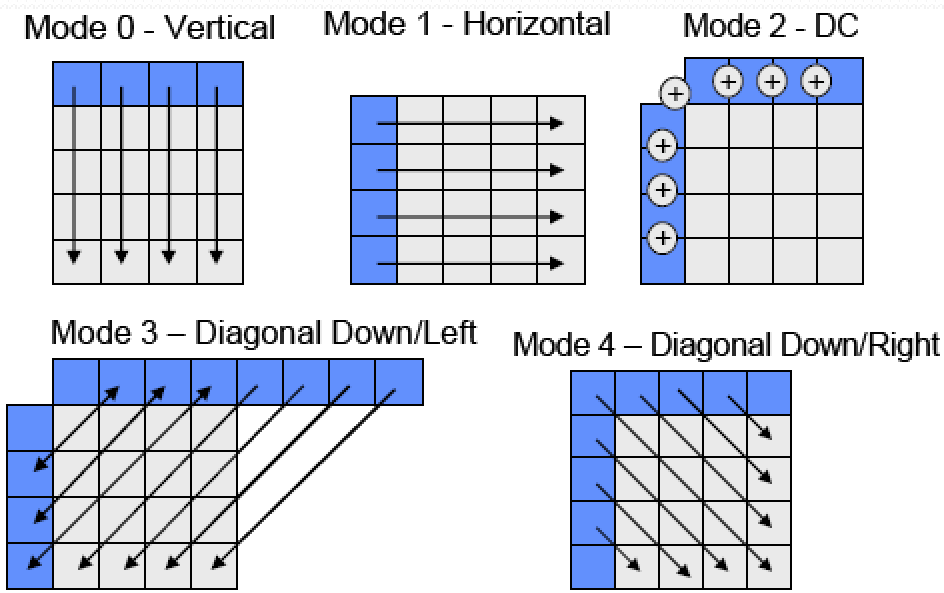
- Reduces blocking artifact
- Operates in-loop therefore more effective than post processing
- Adaptive filtering
- Significant computational complexity
- 5-15% improvement in bitrate per given quality compared to unfiltered video



H.264 Intra Prediction

- Exploit spatial correlation between adjacent blocks in intra frames





H.264 Coding Artifacts

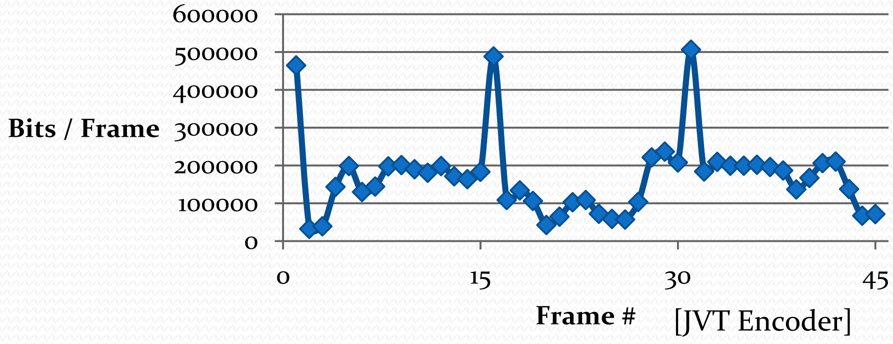
- **Blocking** – reduced due to deblocking filter
- **Ringing** – not observed, due to 4x4 blocks & filter
- **Blurring** – only at very low bitrates
- **Color bleeding** – wasn't observed (4x4 blocks)
- **Temporal discontinuities**
  - Most noticeable artifact in H.264
  - Include a wide range of phenomena. No agreement in the literature on terminology or causes
    Mobile 250Kbps

MPEG2 - Temporal Post Processing Methods

- Delcorso et al., Mosquito Noise Reducer, 2002
- Atzori et al., Adaptive Anisotropic Filter, 2002
- Coudoux et al., Temporal Busyness Post Processor, 2003
- Most literature deals with MPEG2 temporal discontinuities
- Temporal discontinuities characterization not relevant to H.264
  - Practically no ringing in H.264
  - "Temporal discontinuities" is too wide a term

H.264 Flicker

- Noticeable temporal discontinuity around intra frames
- Intra frame may be sharper or more blurred than preceding inter frame
- Intra frame requires different bitrate than inter frames
- Most noticeable in low-medium bitrates



Flicker in the literature

- Fan et al., 2002
  - All-intra video sequences
  - Attribute flicker to changes in intra prediction modes
  - Propose non-compliant encoder modification
  - Referenced measure
    - Compare differences in adjacent frames in encoded and original videos

$$\delta_{orig} = \left| f_I - f_{I-1} \right| \qquad \delta_{encoded} = \left| \hat{f}_I - \hat{f}_{I-1} \right|$$

$$\hat{f}_I = coded\ frame(\#I)$$

$$f_I = original\ frame(\#I)$$

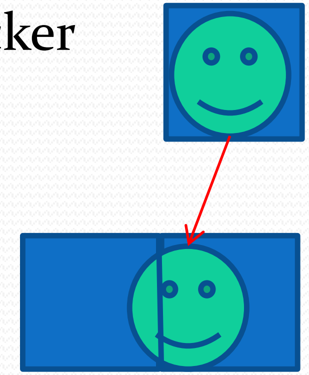
$$flicker = avg \left| \delta_{orig} - \delta_{encoded} \right|$$

Flicker in the literature – cont'd

- Later works treat videos with periodically-inserted intra frames
- Most use encoder modifications:
  - Sakaida et al., 2004 – change intra prediction mode selection, and encode repeatedly with finer quantization
  - Chun et al., 2006 – change intra pred. mode selection
  - Chono et al., 2006 – modify quantization levels
- Yang, Park, Jeon, 2006 – preprocessing by Kalman filter (all intra)
- All works use (roughly) same objective measure

Examination of Flicker Reasons

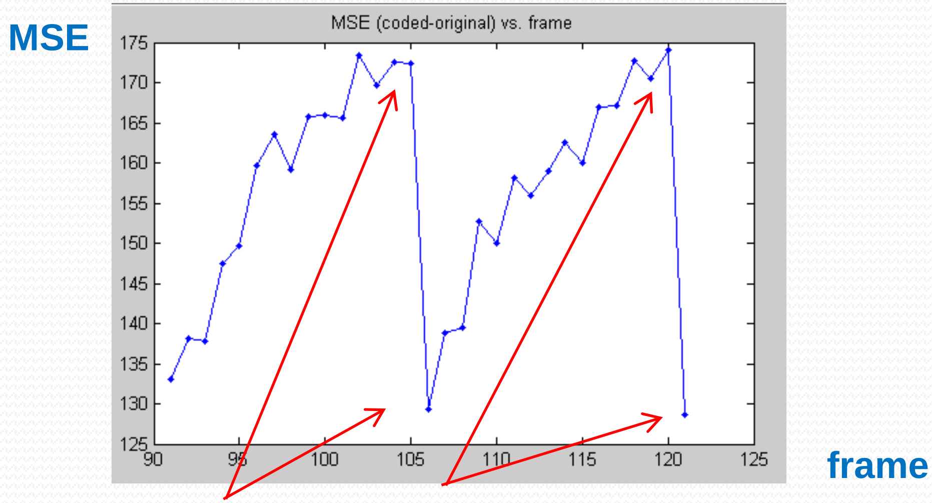
- Different Intra Prediction Modes with SKIP
  - By 4 different papers
  - Not satisfactory because DC-only prediction also exhibits flicker (By DQ – similar test by us, with similar results)
- Grid Movement
  - Objects are broken to different blocks due to movement, each block handled differently, thus flicker is caused
  - Not satisfactory – non moving parts in videos exhibit flicker
- Spirals half with DC modes only



Flicker – Further Examination

- Different Coding Error Patterns (suggested by DQ article)
  - Generalized, includes several components
  - **Inter** – Temporal prediction & strong quantization of residuals or SKIP
  - **Intra** – Spatial prediction & weak quantization of residuals

MSE Coded vs. Original (per pixel)



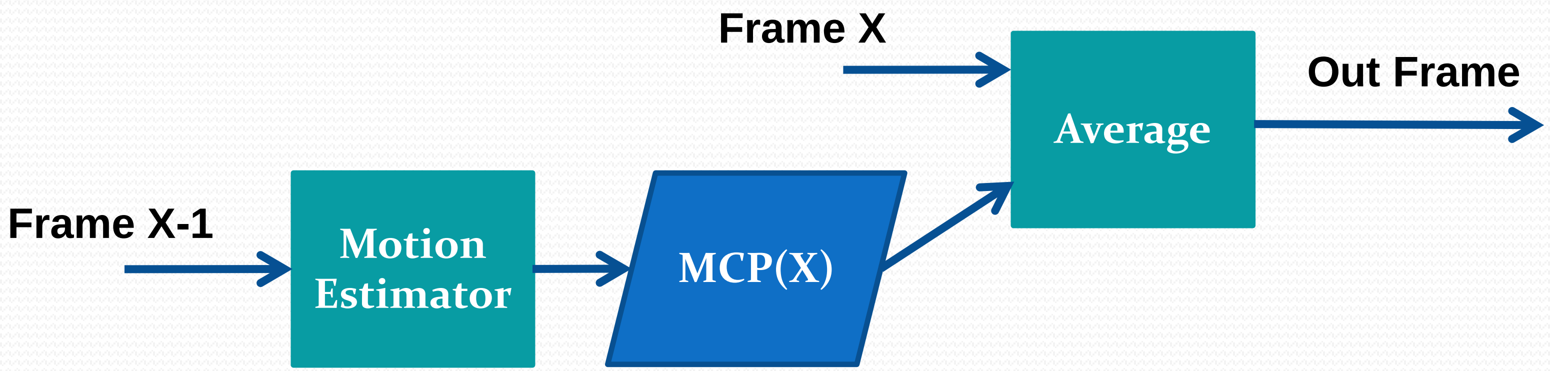
Inter vs. Intra MSE

Flicker Post-Processing

- Novel treatment for flicker
  - Doesn't necessitate changes in the encoder
  - Complements suggested encoder modifications as no single method eliminates flicker completely
  - No Post Processing method for flicker reduction was found in literature
- Difference of Coding Error Patterns
  - **Reduction** – to alleviate flicker
  - **Estimation** – to measure flicker, crucial for adaptive filtering

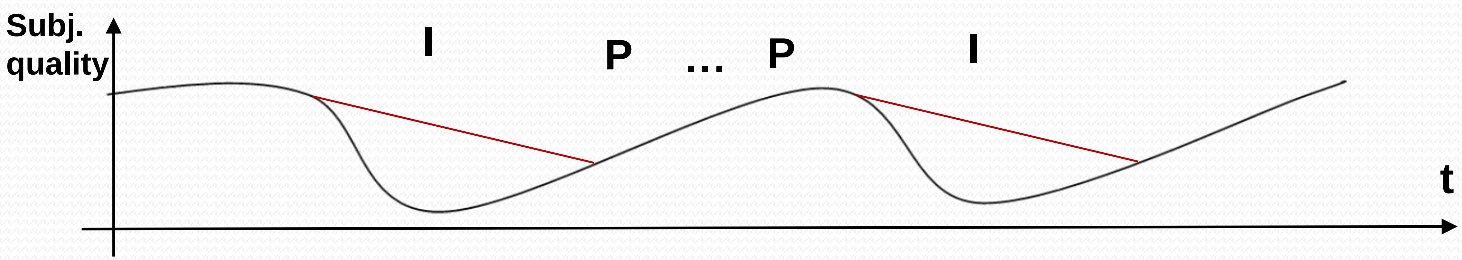
Flicker Reduction - Main Idea

- Estimate motion vectors between every two consecutive frames
- Reconstruct frame X from X-1 by MV, to get MCP(X)
- X is the Intra frame which is Original + spatial prediction error
- MCP(X) **estimates** Original + temporal prediction error
  - The better the estimation the better the results
  - No motion vectors for Intra frames
- Average X and MCP(X)

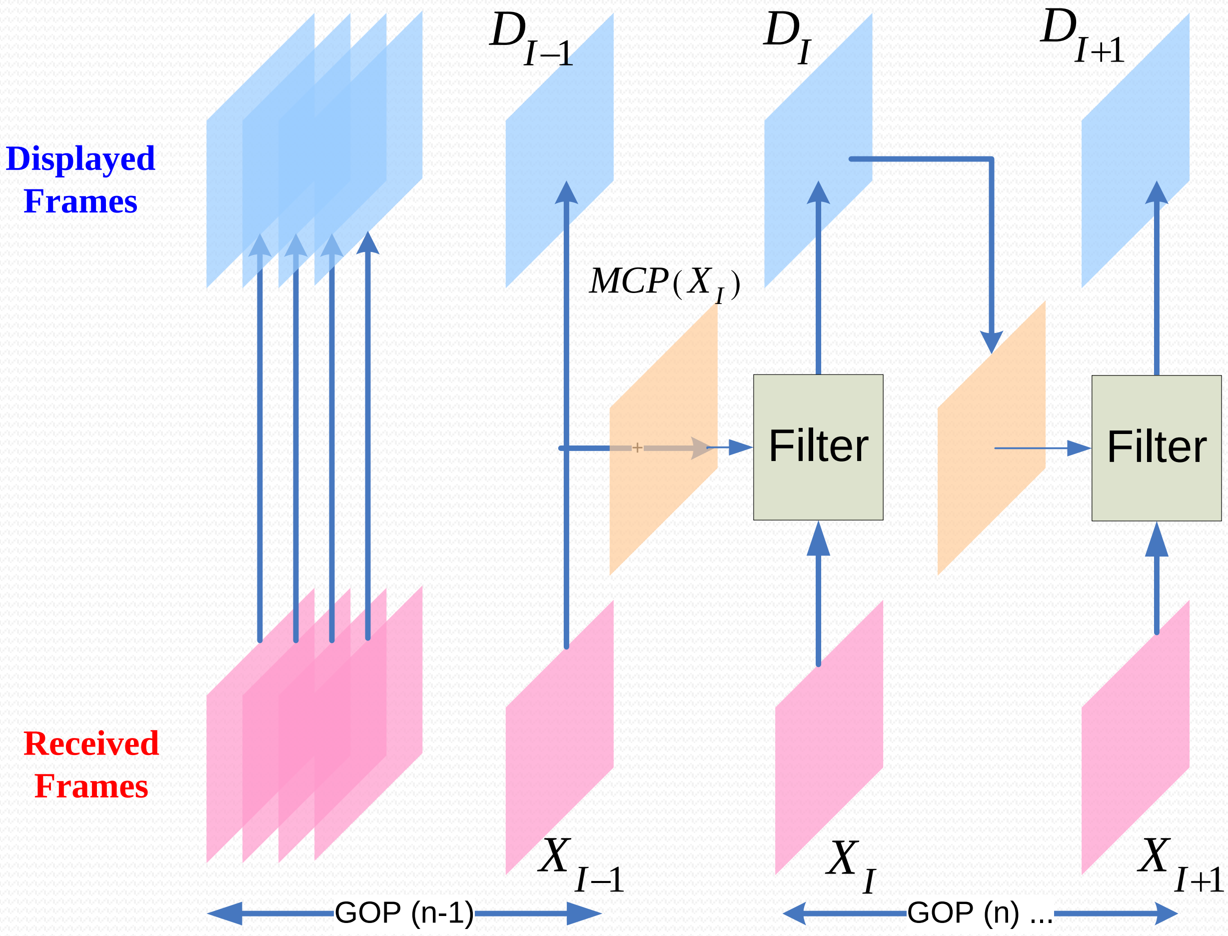


Flicker Reduction – cont'd

- Need to filter only around I frames, to avoid unnecessary blurring
- Jump is steep – need to filter across more than one frame, to smooth the jump
- Use weighted average by distance from I frame

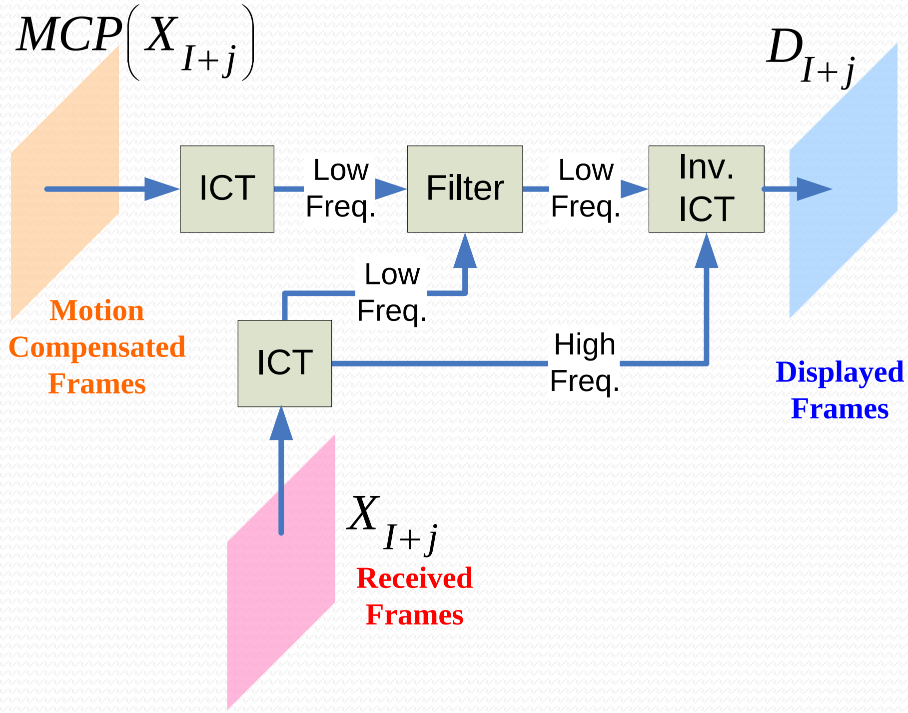


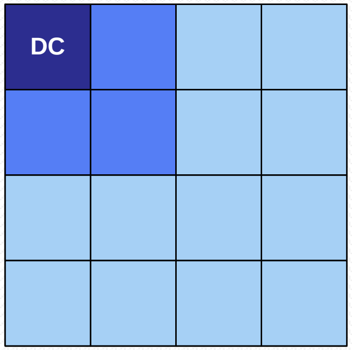
Flicker Reduction Post Processing Scheme



Flicker High Pass Filter

- Use weighted average by distance from I frame
- Only filter low frequencies, to avoid flicker in fine details



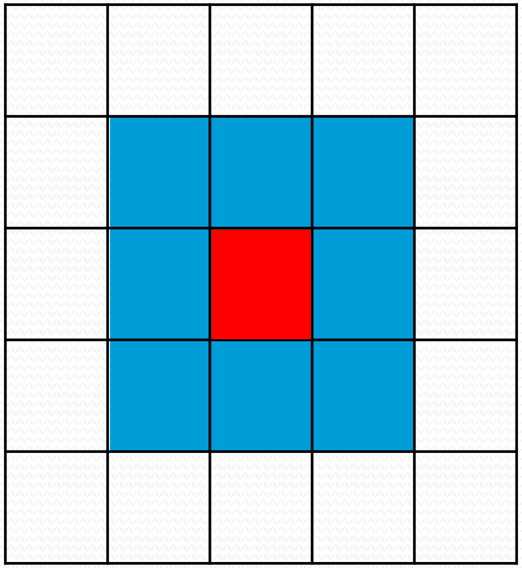


Adaptive Filtering

- Filtering *k* frames reduces intra frame jump by ~1/*k*
- Need to decide how many frames to filter in each GOP
  - Measure flicker in the intra frame
- Earlier works only provide a referenced measure
- Novel non-reference measure was developed, based on empirical flicker characteristics
  - Constructs a ‘flicker map’ for an intra frame using its motion-compensated counterpart
  - ‘flicker map’ is derived from **estimated** difference of coding error patterns (same as X – MCP(X))

Adaptive Filtering – cont'd

- Flicker is more noticeable in smooth areas
- Identify 'smoothness' by calculating 1/(1+std) of 3x3 block centered on the pixel
- 1/(1+std) < 0.5 means a pixel is in a non-smooth area



Adaptive Filtering – cont'd

- Identification of smooth areas

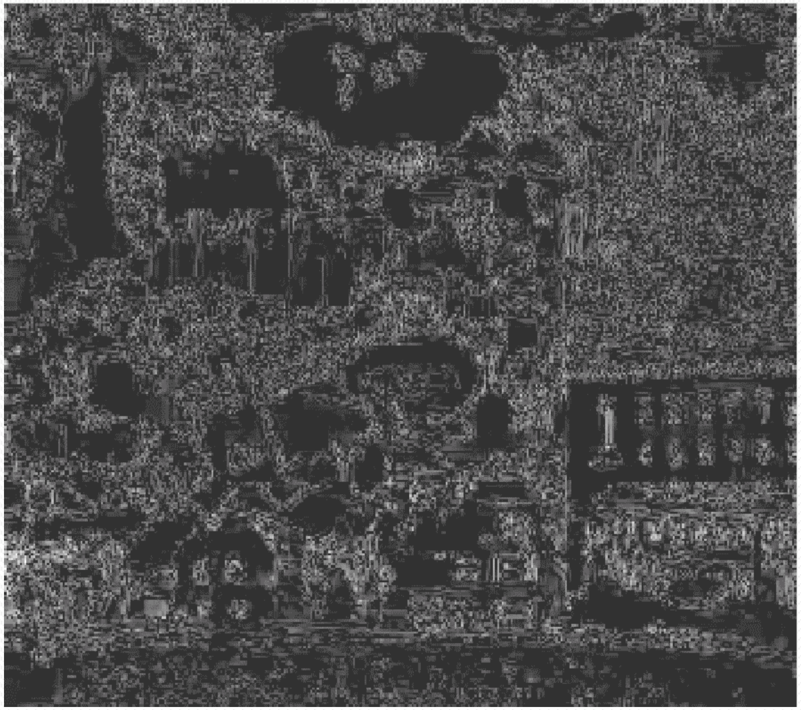
Intra frame

Smooth areas

Adaptive Filtering – cont'd

- Large differences between intra frame and motion-compensated counterpart may indicate high flicker

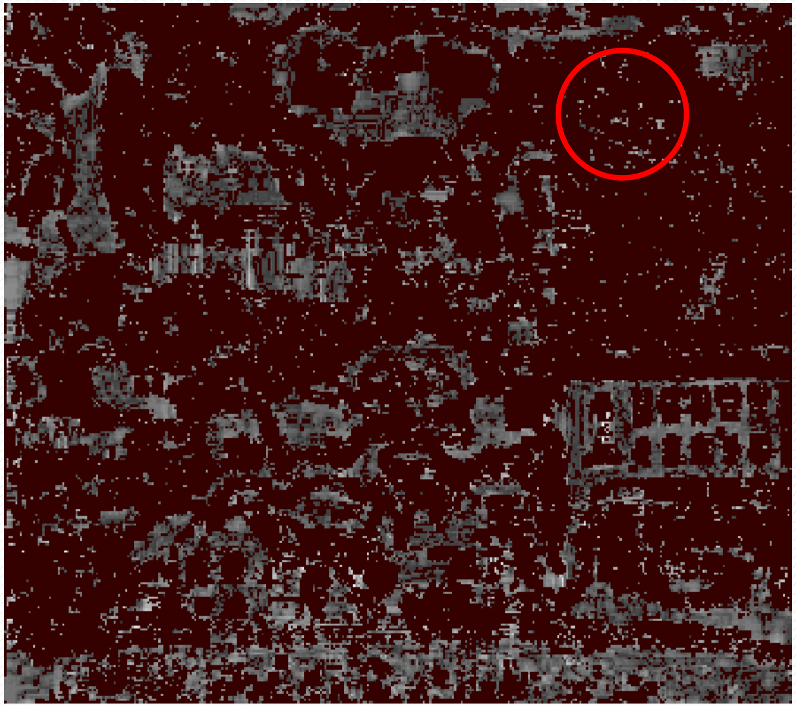
Intra frame



Absolute difference

Adaptive Filtering – cont'd

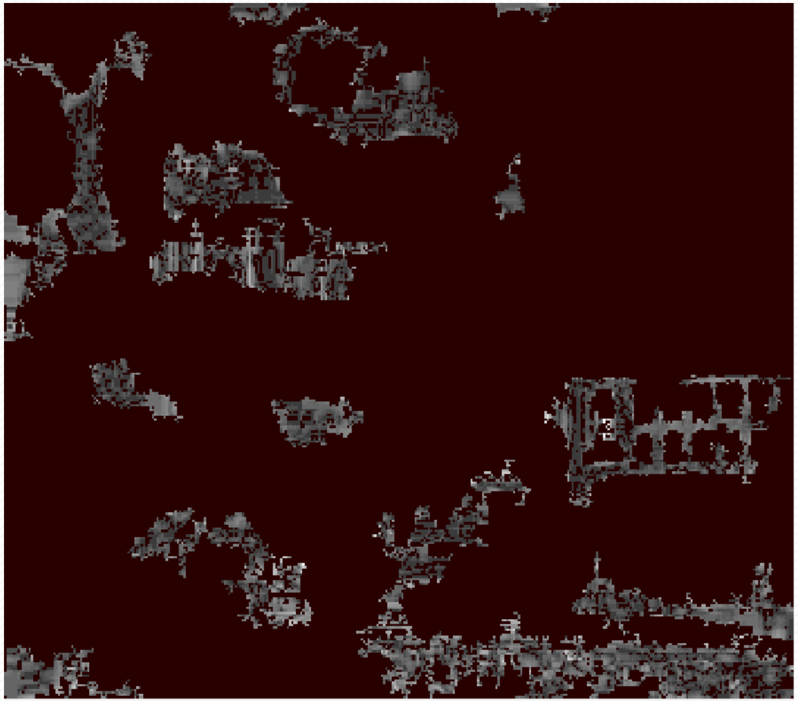
- Smooth areas with large differences relative to motion-compensated image will display most flicker
- Multiply smoothness map by difference map



Abs. diff * smoothness

Adaptive Filtering – cont'd

- Isolated changing pixels are not perceived as flicker
- Use morphological opening by reconstruction to detect clusters of pixels
- Where the result is not zero, copy pixel from difference image

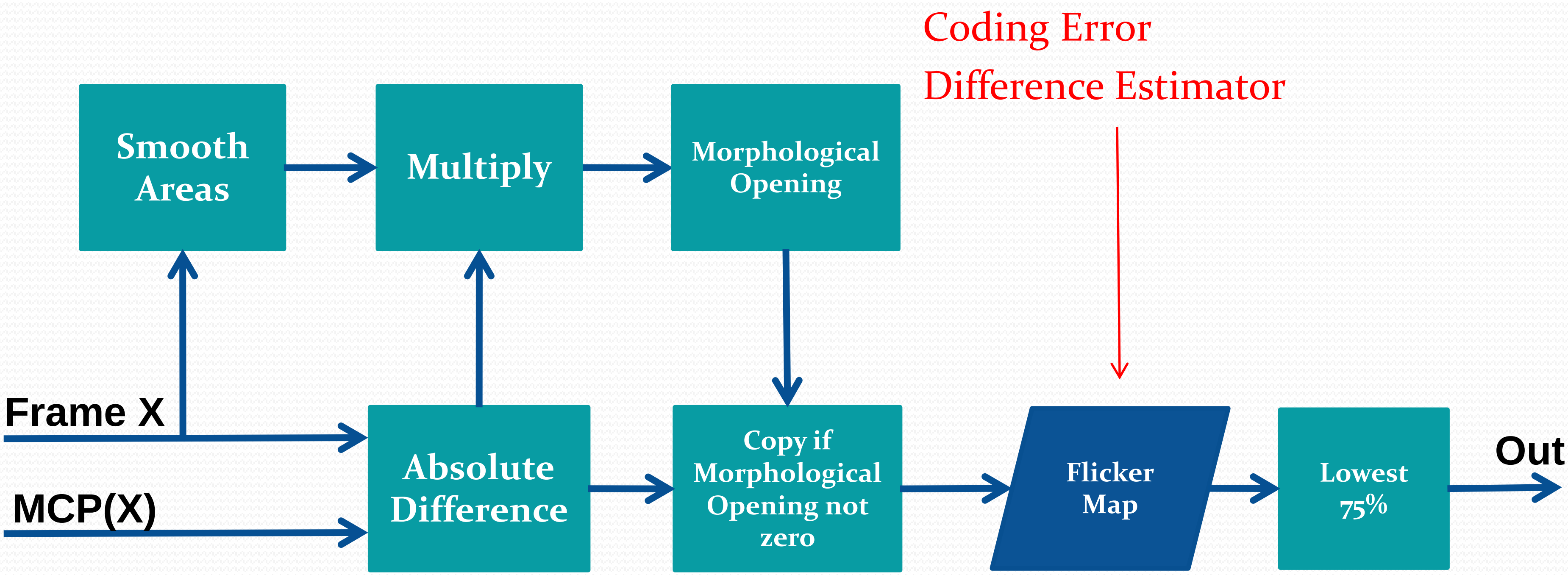


Final flicker map

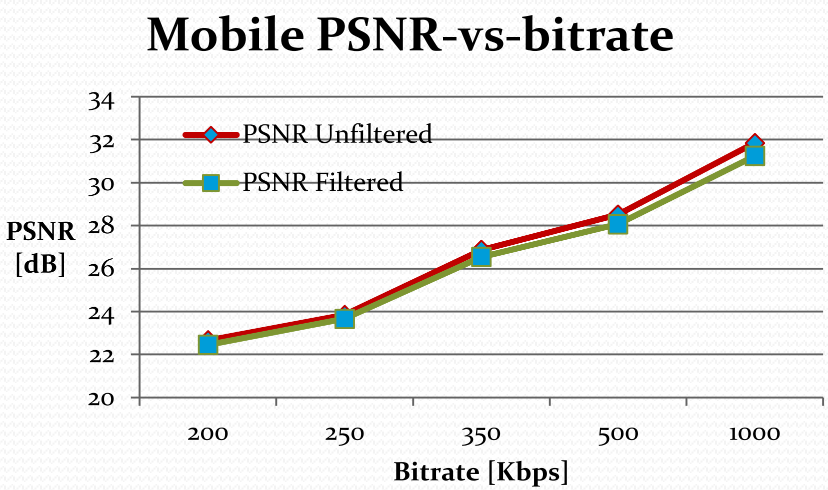
Adaptive Filtering – cont'd

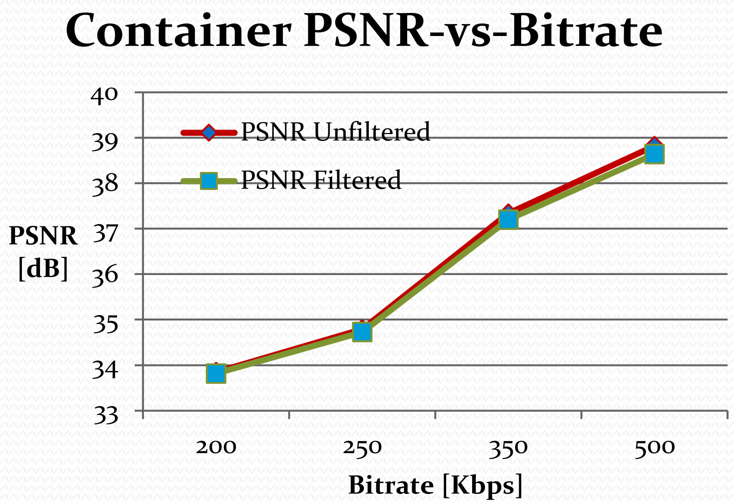
- Flicker map indicates presence and strength of flicker
- Flicker is measured for the entire frame
  - Need to determine the worst flicker, not the average
- Pick lowest integer that is greater than 75% of the non-zero pixels in the flicker map
- Indicates the number of frames to filter in the GOP

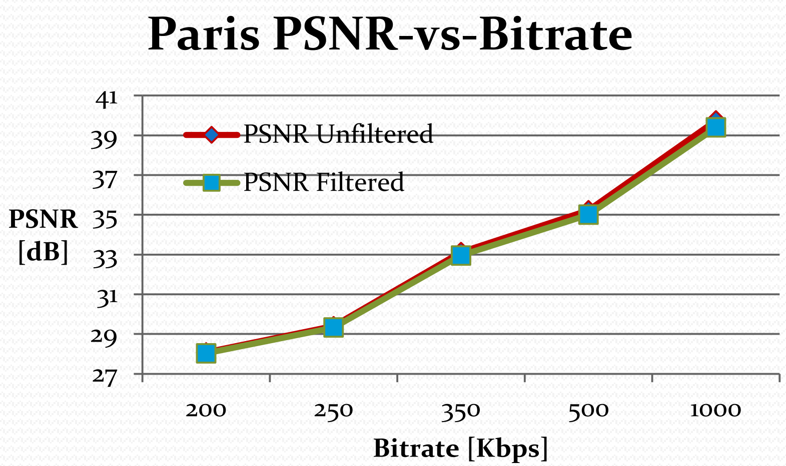
Adaptive Filtering - Diagram



PSNR vs. Bitrate

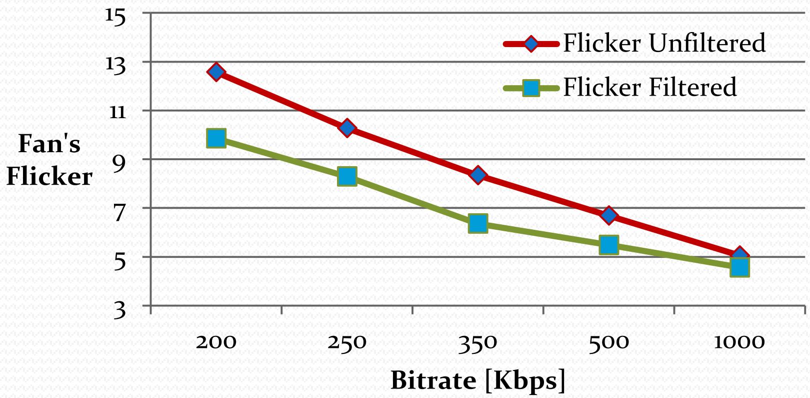




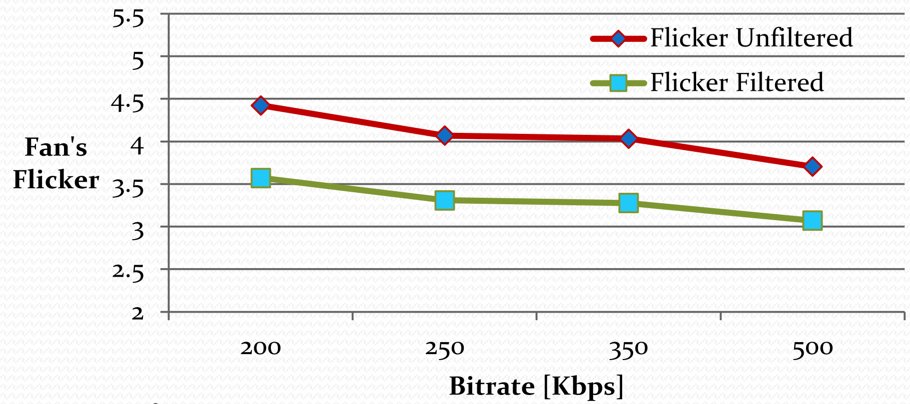


Flicker vs. Bitrate

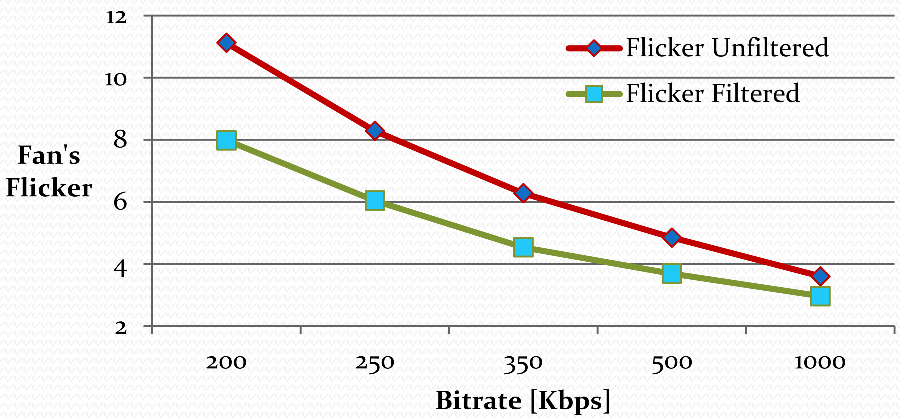
Mobile Flicker-vs-Bitrate



Container Flicker-vs-Bitrate



Paris Flicker-vs-Bitrate



Summary

- Flicker is prevalent in H.264
  - Wasn't studied extensively in the past
  - Existing solutions require encoder changes
- Innovative post processing technique and non-reference objective measure suggested
  - Complements encoder modifications
  - Shows good results, objectively and subjectively
  - Paper submitted to PCS 2007
  - Patent-pending by Intel-Oplus

Future Directions

- Better frame reconstruction
  - Results affected by motion vector accuracy
- Adaptive filtering in the frame
  - Use flicker map to select areas where flicker is particularly noticeable
  - Might lead to edge artifacts

Thank you!

Backup

Coding Artifacts - examples

Color bleeding

Ringing /
Mosquito noise

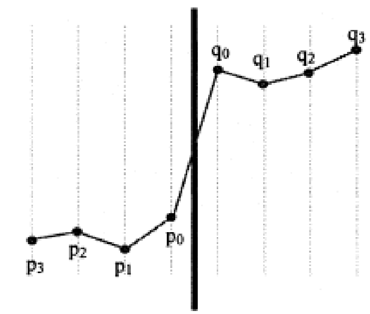
Coding Artifacts – examples

Blurring

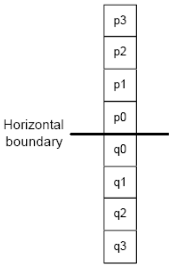
Blocking

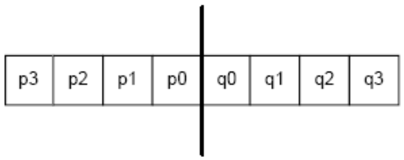
Deblocking Filter

Blockiness across a 4x4 block boundary:

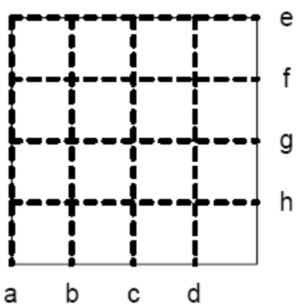


Adaptive filtering along horizontal and vertical edges:

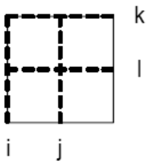




Filter decisions based on block type and position in macroblock:



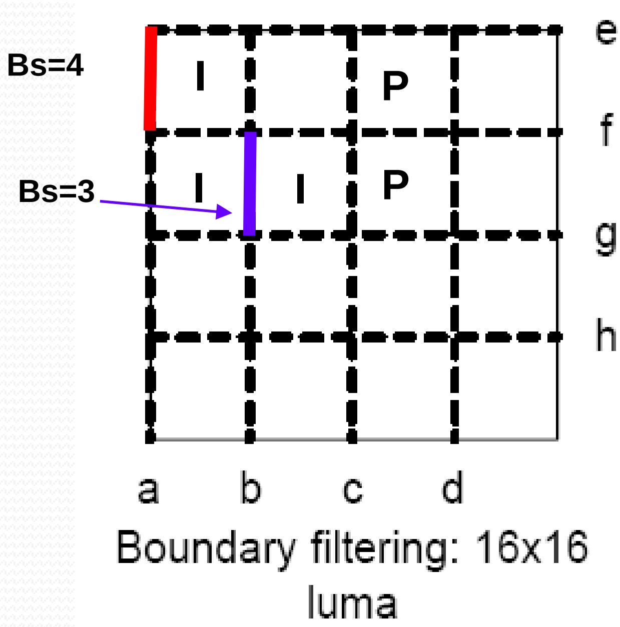
Boundary filtering: 16x16 luma



8x8 chroma

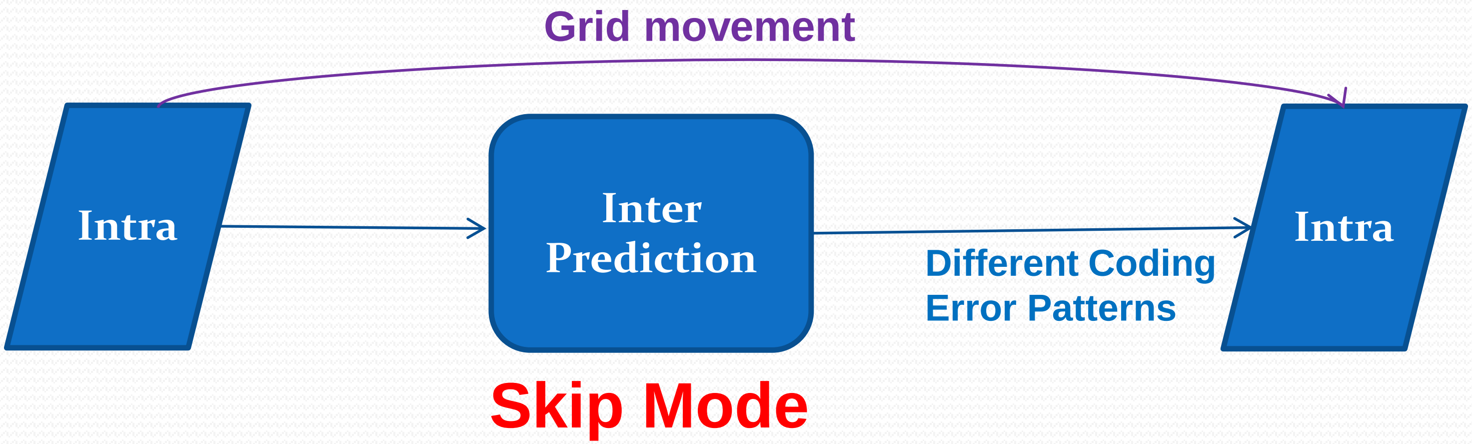
Deblocking Filter

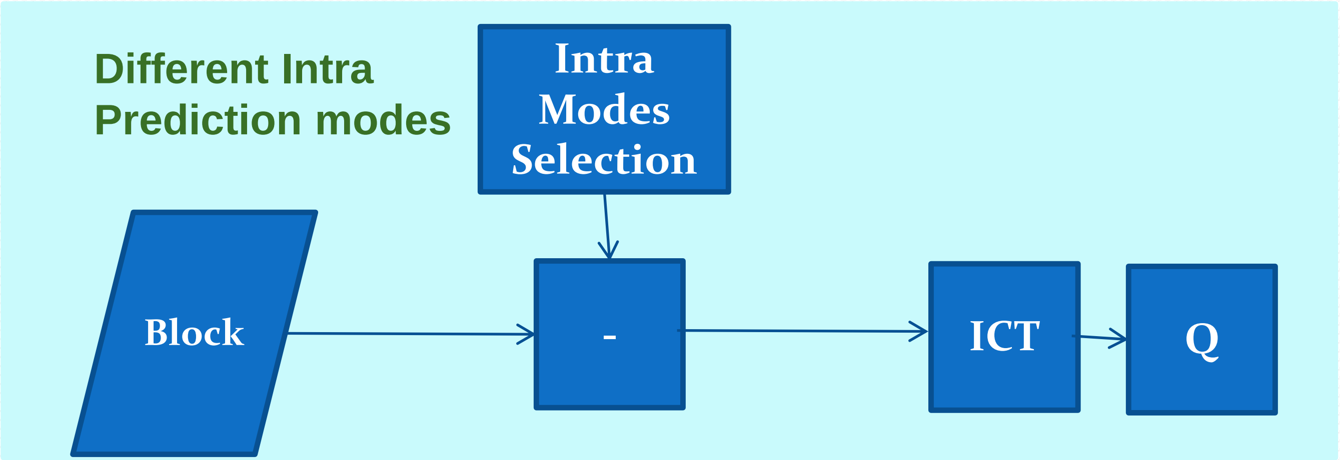
- Filter Strength can be altered on the:
  - Slice level
  - Macroblock level
  - Sample level



Block modes and conditions	Bs
One of the blocks is Intra *and* the edge is a macroblock edge	4
One of the blocks is Intra	3
One of the blocks has coded residuals	2
Difference of block motion $\geq$ 1 luma sample distance	1
Motion compensation from different reference frames	1
Else	0

Flicker – Suggested Reasons





Backup2

References

[1] Sandra Delcorso, Carolina Miro, and Joel Jung **MNR: A Novel Approach to Correct MPEG Temporal Distortions**

[2] Atzori, L.; De Natale, F.G.B.; Granelli, F **Adaptive anisotropic filtering (AAF) for real-time visualenhancement of MPEG-coded video sequences**.

[3] Francois-Xavier Coudoux, Marc Georges Gazalet, Patrick Corlay **A post-processor for reducing temporal busyness in low-bit-rate video applications**

Blurring

- Experienced at low-medium bitrates
  - Happens due to low bitrate and due to the de-blocking filter. Annoying blocking artifact is replaced by less annoying blurring
  - Some details were simply lost (due to bitrate)
  - It is not clear that something can be done about it

Temporal Busyness Post Processor

- Coudoux, Gazalet, Corlay, 2003
- Deals with temporal busyness resulting from ringing and DCT basis images
  - DCT basis images not present in H.264 due to deblocking filter
  - Ringing is not a problem in H.264

MNR – Drawbacks

- Simple motion identification:
  - Doesn’t use motion vectors
  - Uses only absolute difference between same blocks in adjacent frames
- Weak filter:
  - Filters only DC coefficients
  - Uses only 2 frames for filtering (preceding and following)
- No perceivable improvement in our videos.

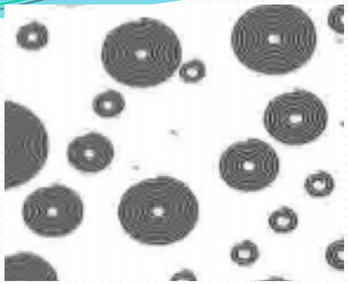
AAF – Drawbacks

- Doesn't do temporal filtering
  - Assumes that mosquito noise comes only from ringing
  - Doesn't mention other temporal artifacts besides "mosquito noise"
  - There are other temporal artifacts

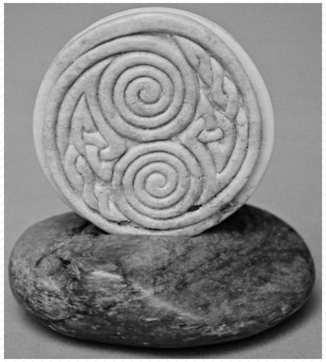
H.264 Encoders

- A variety of H.264 encoders in the market
- x264 is the leading encoder according to benchmarks (MSU)
  - Chosen encoder for project
- The JVT reference encoder is considerably inferior
  - Also exhibits motion jerkiness (at low bitrates)

Artificial Example: Spirals

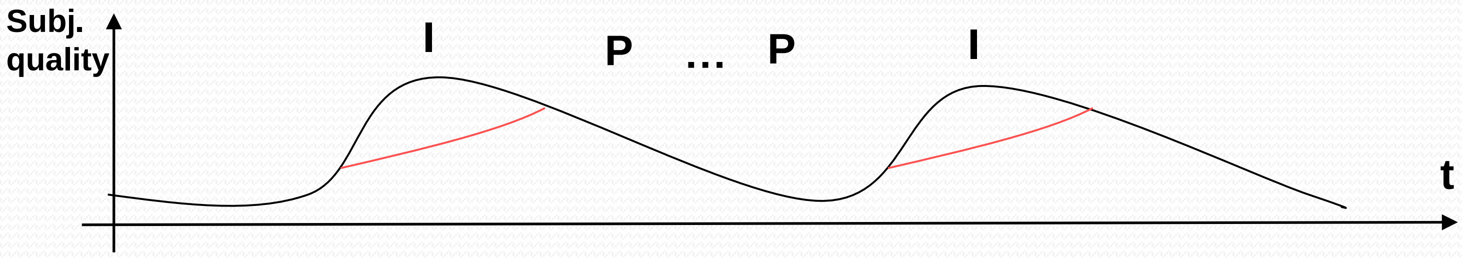


- Based on Fenimore, Libert, Roitman, 2000
  - Propose a metric for MPEG2 MN measurement
  - Propose a test pattern for subjective MN measurement: still spirals video
- We used a similar pattern (800x530x64Kbps)
  - Still video exhibits slight PI Jumps
  - Much worse jumps with movement
  - <Moving video example>



From Artificial to Real World Video

- Need to filter only around I frames, to avoid unnecessary blurring
- PI Jump is steep – need to filter across more than one frame
- Use weighted average by distance from I frame



From Artificial to Real World Video – cont.

- I frame doesn't have motion vectors
  - So we don't use them…
  - We generated our own MVs from the original video, and used them in the reconstruction
  - In real applications, can use H.264 MVs, and generate I frame MVs by motion estimation with MSE

From Artificial to Real World Video – cont.

- Motion estimation is not perfect
- Filtering high frequencies (=textures and edges) will cause an edge jump when we stop filtering
- Solution: filter only low frequencies

Approach Summary

- Generate MVs for entire original video
- fout(Intra-1)=f(Intra-1) % don't process pre-I frames
- For each frame j in frames: **Intra** to **Intra+k-1**
  - fc(j) = compensate_motion{fout(j-1)}
  - fout(j)= low_freq {j/k*f(j) + (k-j)/k*fc(j)}+high_freq{f(j)}

Video Examples

- Mobile, unfiltered
- Mobile, filter low frequencies
- Shields, unfiltered
- Shields, filter all frequencies
- Shields, filter low frequencies
- Ballroom, unfiltered
- Ballroom, filter low frequencies

Future Directions

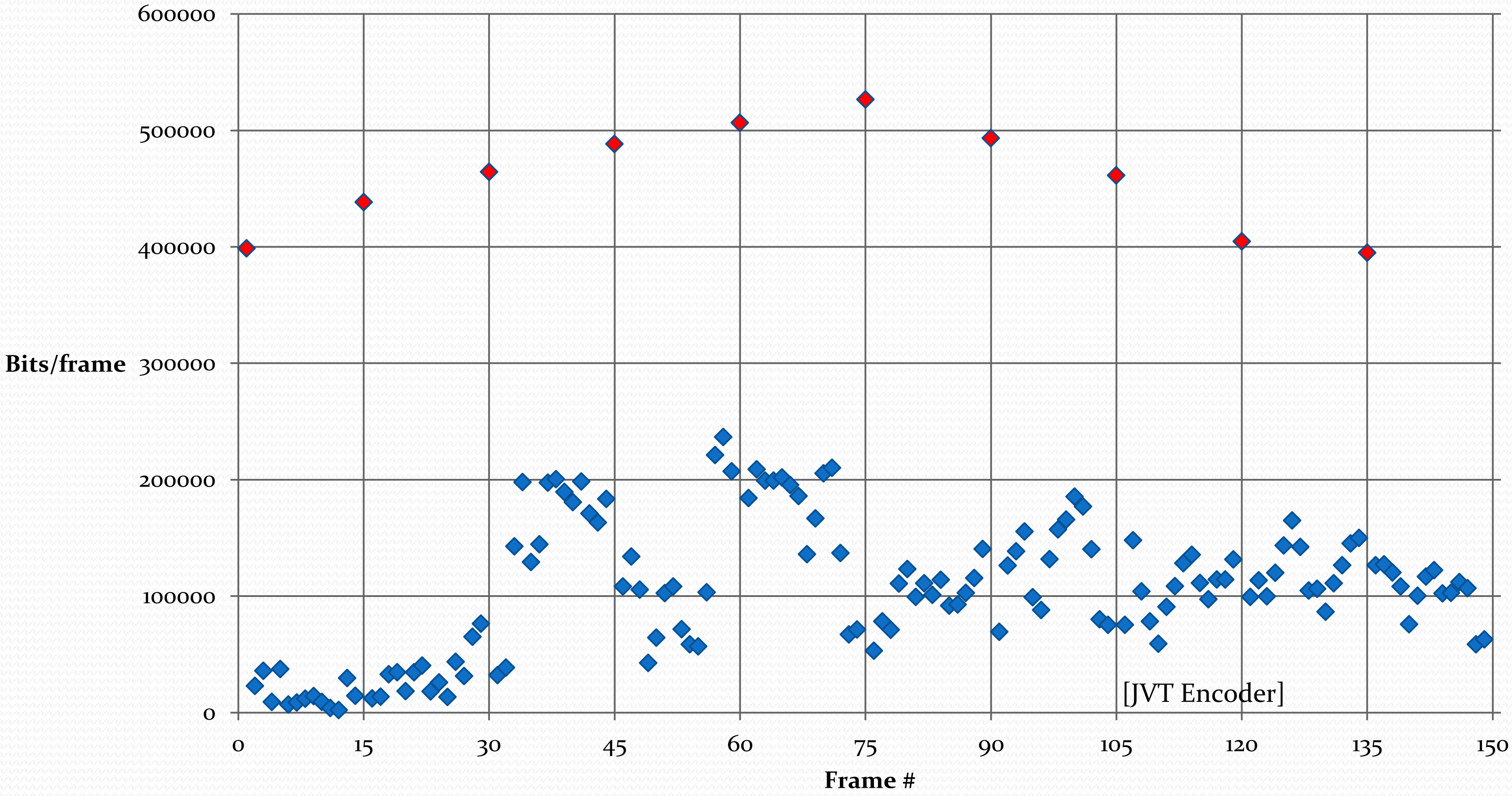
- I frame motion vectors:
  - Generate by exhaustive search
  - Interpolate I-1 and I+1 MVs
- Objective quality metric
- Optimal thresholds

MNR – Mosquito noise reducer

- Delcorso, Jung, 2002
- Defines Mosquito Noise as temporal fluctuation near edges of moving objects
  - Identifies moving blocks (LPF on frame difference)
  - DC median filter (temporal & spatial) on still blocks
- Drawbacks
  - Simple motion estimation
  - Weak filter
  - No perceivable improvement in our videos

AAF – Adaptive Anisotropic Filter

- Atzori, De Natale, Granelli, 2002
- Defines Mosquito Noise as ringing near edges of objects
  - Identifies the types of blocks
  - Applies a set of spatial filters on different types of blocks
- Drawbacks
  - No temporal filtering
  - Not all temporal artifacts are due to ringing


Bits/frame
600000
500000
400000
300000
200000
100000
0
0
15
30
45
60
75
90
105
120
135
150
Frame #
[JVT Encoder]

Objective Measurement Results

- Modified version of Fan's flicker measure
  - Apply measure only to k frames following I-frame (k=2, 4, 6)
  - For our filter, low k values expected to give better results

	Distortion Reduction		
Video	**2 frames**	**4 frames**	**6 frames**
mobile175i15	58.67%	46.16%	38.83%
ballroom300i15	38.25%	25.05%	-
shields512i15	22.50%	17.86%	-

Adaptive Filtering - Summary

- Locate smooth areas in image
- Calculate absolute difference between intra frame and motion-compensated counterpart
- Multiply images and do opening by reconstruction
- Where the result is not zero, retain values from absolute difference image
- Determine strength of flicker in the resulting image
- **Draw Instead!!!**

Modified Flicker Measure

- Fan's flicker measure designed for all-intra videos
  - Averages flicker for entire video
  - Flicker is only noticeable around intra frames
  - When using periodically-inserted intra frames, averaging over the entire video will mask the jump
- For filtered videos, flicker was averaged only for first 6 frames of each GOP

No Filter vs. Filter

Temporal Post Processing Methods

- Mosquito Noise Reducer: 2002, [1]
  - Identifies moving blocks (LPF on frame difference)
  - DC median filter (temporal & spatial) on still blocks
  - No perceivable results on H.264 videos
- Adaptive Anisotropic Filter: 2002, [2]
  - Identifies the types of blocks
  - Applies a set of spatial filters on different block types
  - Deals with ringing-related noise, not relevant to H.264
- Others (e.g. [3]) - Similar drawbacks

Intermediate Conclusions

- Most literature deals with MPEG2 temporal discontinuities
- Temporal discontinuities characterization not relevant to H.264
  - Practically no ringing in H.264
  - “Temporal discontinuities” is too wide a term

PSNR & Flicker vs. Bitrate