Distributed Video Coding using the Wavelet Transform

Yair Carmon and Alona Strugatzki
Supervisor: Yevgeny Priziment
August, 2008
The Problem

- Mainstream video encoders are complex and therefore expensive (in time, money, power or all of the above)
- The reason: Motion estimation is has high complexity
- The resulting paradigm:
A More Desirable Paradigm

Many cheap encoders

One expensive decoder
System outline

- Wavelet transform: 9/7 biorthogonal Daubechies filters
- Motion classification?
Modular Reduction

- Wavelet coefficients passed through folding function with

\[ X_{MR} = MR(X, M) = X - M \left( \frac{X \bmod M}{M} \right) \]

- Code length depends mainly on the # of bit planes after MR:

\[ M = 2, 6, 14, 30, 62, 126, 254, 1022, \ldots \]
Reconstruction

- Each coefficient reconstructed separately based on SI and “CS”
- Correlation model: \( p_{x|s}(x) = \frac{1}{2a} \exp\left(-\frac{|x - s|}{a}\right) \)
- SPIHT quantization error: \( \Delta \)
- Reconstruction process:
Remarks and conclusions

- The system from [1] was implemented and investigated
- Single frame performance was better than in the article
- The parameter assignment scheme was “reverse engineered”
- Best performance achieved with extremely nonuniform bitrate allocation
- Requires large buffers – not feasible in every coding scenario
Motivation for localized parameters

- Many frames have “background” and “foreground” regions
- Good SI in the background and bad SI in the foreground
- Wavelet Transform is localized
- Use different M for different blocks (in wavelet domain)
Motivation – cont’

$M = 1022$
$M = 510$
$M = 254$
$M = 126$

Side Info
PNSR = 28.47

M = 126
PNSR = 29.88

M = 254
PNSR = 31.7

M = 510
PNSR = 33.34

M = 1022
PNSR = 33.91

Variable M
PNSR = 34.82
Performance and Remarks

- System has only 2 free parameters: SM and κ.
- Performance close to upper bound for constant bitrate:

No option to allocate a variable bitrate per frame.
Flexible bit allocation

- $n_{\text{min}}$ controls the quantization level
- “carphone”
- 300 Kbps
- $n_{\text{min}} = 3$
- GOWZ size=8
Performance

![Graph showing performance comparison between different generations and learned thresholds for 'foreman' and 'carphone' datasets.](image-url)
Performance Comparison

![Graph showing PSNR vs rate for 'foreman' and 'carphone' datasets. The graph compares 'Our system' and 'DISCOVER'.]
Quality distribution

Temporal distortion distribution - foreman

Proposed

DISCOVER

PSNR [dB]

Frame #

150 Kbps
400 Kbps

Foreman

Frame Number

PSNR [dB]

Frame Number

150 Kbps
400 Kbps
## Comparison

<table>
<thead>
<tr>
<th></th>
<th><strong>Our system</strong></th>
<th><strong>DISCOVER</strong></th>
</tr>
</thead>
</table>
| Rate Control                | • J2K – direct intra rate control  
• Control of average WZ rate  
• \( N_{\text{GOWZ}} \) controls WZ rate deviation | Only indirect control of bitrate through intra & WZ quantization step sizes |
| Feedback Channel            | Does not use a feedback channel                                               | Uses a low rate feedback channel                                             |
| Adaptivity                  | • \( M, n_{\text{min}} \) determined adaptively using \( D \)    
• \( n_{\text{min}} \) allocation learned offline                      | • Intra/WZ Q-param matching offline  
• Adaptive WZ rate control via feedback                                  |
| Channel Coding              | Scalar “Continuous syndrome”, quantized with SPIHT                              | Uses LDPC syndrome                                                            |
| Performance                 | • DISCOVER has better RD performance in terms of PSNR, and in most cases visually 
• Temporal distortion distribution is similar in both systems              |                                                                              |