Optimization Methods for Data Compression

Giovanni Motta
Motivations

Data compression algorithms use heuristics

• How good a given heuristic is?
• What if an heuristic is replaced by an optimal procedure?
• Can we simplify an optimal procedure to derive a good heuristic?
• How compressible is a given data set?
Relevance

• Insight about best obtainable performance of a class of algorithms
• Validation of the old and derivation of new heuristics
• Better estimation of empirical entropy

Answers are given case-by-case
Contributions

- New algorithms proposed and studied
- Improvements of state of the art algorithms
- New heuristics derived from the optimal case
- Derive upper bounds on compression of standard data sets
Areas of research

- **Lossy**
  - Trellis Coded Vector Quantization
  - H.263+ frame skipping optimization
  - JPEG domain processing
- **Lossless**
  - Linear Prediction and classification
  - Polynomial texture maps
  - Discrete colors images
Areas of research

- **Lossy**
  - Trellis Coded Vector Quantization
  - H.263+ frame skipping optimization
  - JPEG domain processing
- **Lossless**
  - Linear Prediction and classification
  - Polynomial texture maps
  - Discrete colors images
Areas of research

• Lossy
  • Trellis Coded Vector Quantization
  • H.263+ frame skipping optimization
  • JPEG domain processing
• Lossless
  • Linear Prediction and Classification
  • Polynomial texture maps
  • Discrete colors images
Vector Quantization

- Most general source coding method
- Lossy
- Asymptotically optimal - probabilistic proof
- Time and space complexity grow exponentially with vector dimension
- Codebook design is NP complete
- LBG designed ESVQ provides upper bound on practical performance
Trellis Coded Residual VQ

- New VQ proposed by us
- Combines residual quantization and trellis coding
- Optimal or greedy codebook design
- Viterbi search
- Good performance both on image and LP speech coding
- Suitable for progressive encoding
Residual Quantization

\[ Q_0(x_0) \rightarrow \hat{x}_0 \rightarrow Q_1(x_1) \rightarrow \hat{x}_1 \rightarrow x_2 \]
Trellis Coded Residual VQ
Trellis Coded Residual VQ

Vector Quantizer

- Codebook
  \[ A = \{ y_1, y_2, \ldots, y_N \} \]
- Partitions
  \[ P = \{ S_1, S_2, \ldots, S_N \} \]
- Mapping
  \[ Q(x) = y_j \text{ if and only if } x \in S_j \]
Trellis Coded Residual VQ

Coding distortion depends on \( F_{x^1, \ldots, x^P} \)

\[
D(x^1, \hat{x}^1) = \int \cdots \int d \left[ x^1, \sum_{p=1}^{P} Q^p(x^p) \right] dF_{x^1, \ldots, x^P}
\]

Direct Sum (or Equivalent) Quantizer

\[
D(x^1, \hat{x}^1) = \int d \left[ x^1, Q^e(x^1) \right] dF_{x^1}
\]
Trellis Coded Residual VQ

- Optimality conditions derived for the code vectors
- Partitions must be Voronoi. Too expensive to specify partition boundaries
- Optimality conditions apply to full search
- Sequential codebook design (greedy)
- Viterbi search
Trellis Coded Residual VQ

• Viterbi search algorithm (shortest path)
# Trellis Coded Residual VQ

Random Sources

<table>
<thead>
<tr>
<th>bps</th>
<th>Uniform</th>
<th>Gauss</th>
<th>Laplace</th>
<th>Bimodal</th>
<th>Markov</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.3</td>
<td>1.24</td>
<td>1.17</td>
<td>1.18</td>
<td>2.87</td>
<td>4.40</td>
</tr>
<tr>
<td>0.6</td>
<td>2.82</td>
<td>2.54</td>
<td>2.63</td>
<td>5.41</td>
<td>7.77</td>
</tr>
<tr>
<td>0.9</td>
<td>4.64</td>
<td>4.04</td>
<td>4.19</td>
<td>7.46</td>
<td>10.48</td>
</tr>
<tr>
<td>1.2</td>
<td>6.26</td>
<td>5.55</td>
<td>5.57</td>
<td>8.86</td>
<td>12.77</td>
</tr>
<tr>
<td>1.5</td>
<td>7.82</td>
<td>7.11</td>
<td>7.02</td>
<td>10.26</td>
<td>14.72</td>
</tr>
<tr>
<td>1.8</td>
<td>9.44</td>
<td>8.75</td>
<td>8.53</td>
<td>11.74</td>
<td>16.49</td>
</tr>
<tr>
<td>2.1</td>
<td>11.08</td>
<td>10.35</td>
<td>10.01</td>
<td>13.10</td>
<td>18.19</td>
</tr>
<tr>
<td>2.4</td>
<td>12.77</td>
<td>11.98</td>
<td>11.48</td>
<td>14.51</td>
<td>19.61</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>bps</th>
<th>Uniform</th>
<th>Gauss</th>
<th>Laplace</th>
<th>Bimodal</th>
<th>Markov</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.3</td>
<td>1.24</td>
<td>1.17</td>
<td>1.18</td>
<td>2.20</td>
<td>4.40</td>
</tr>
<tr>
<td>0.6</td>
<td>2.80</td>
<td>2.52</td>
<td>2.58</td>
<td>4.16</td>
<td>7.63</td>
</tr>
<tr>
<td>0.9</td>
<td>4.56</td>
<td>3.95</td>
<td>4.03</td>
<td>6.50</td>
<td>10.19</td>
</tr>
<tr>
<td>1.2</td>
<td>5.89</td>
<td>5.19</td>
<td>5.21</td>
<td>7.80</td>
<td>12.20</td>
</tr>
<tr>
<td>1.5</td>
<td>7.23</td>
<td>6.49</td>
<td>6.43</td>
<td>9.03</td>
<td>13.98</td>
</tr>
<tr>
<td>1.8</td>
<td>8.70</td>
<td>7.84</td>
<td>7.67</td>
<td>10.32</td>
<td>15.51</td>
</tr>
<tr>
<td>2.1</td>
<td>10.00</td>
<td>9.11</td>
<td>8.85</td>
<td>11.46</td>
<td>16.86</td>
</tr>
<tr>
<td>2.4</td>
<td>11.31</td>
<td>10.38</td>
<td>10.05</td>
<td>12.57</td>
<td>18.10</td>
</tr>
</tbody>
</table>

Full Search

Viterbi

<table>
<thead>
<tr>
<th>bps</th>
<th>Uniform</th>
<th>Gauss</th>
<th>Laplace</th>
<th>Bimodal</th>
<th>Markov</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.9</td>
<td>5.59</td>
<td>4.74</td>
<td>5.04</td>
<td>7.64</td>
<td>11.47</td>
</tr>
<tr>
<td>1.2</td>
<td>8.13</td>
<td>7.20</td>
<td>7.47</td>
<td>10.27</td>
<td>14.62</td>
</tr>
</tbody>
</table>

ESVQ
Trellis Coded Residual VQ

Gauss-Markov Random Source

SNR (dB)

bits per sample

Viterbi
Full Search
ESVQ
Trellis Coded Residual VQ

- Training and Test set images (12+16):
  - 512x512 pixels
  - 256 gray levels
  - vectors of 3x3 and 4x4 pixels each

- Error measured in SQNR

- Compared with:
  - LBG designed ESVQ
  - Goldschneider's VQ package (fixed and variable rate tree quantizers and an ESVQ that uses the code book of the tree quantizers).
Trellis Coded Residual VQ

Grey level Image Coding

TCVRQ vs. ESVQ performance on test set images
Trellis Coded Residual VQ

Grey level Image Coding

TCVRQ vs. tree VQ performance on test set images
Trellis Coded Residual VQ

- Tests performed on a 2.4 Kbit/sec Linear Prediction based codec
- Quantization of the LP parameters is critical
- LP parameters represented as Line Spectrum Frequencies (or LSFs)
- LSFs quantized at bit-rates of 1.9-2.4 bits per parameter with a 10-stages TCVRQ.
Trellis Coded Residual VQ

- Speech
- LP Analysis (Burg Method)
- V/UV Decision
  - Pitch Period
  - Stochastic Codebook
  - Single-Pulse Generator
  - Error Estimation
- V/UV Decision
- Synthesis
  - LP Parameters
  - Trellis Coded Vector Residual Quantizer

Excitation

V/UV
Trellis Coded Residual VQ

- Training set: 76 sentences, male and female speakers, most common European languages

- Test set: English sentences, male and female speakers. Phonetically rich and hard to encode sentences

- Error measured in terms of Cepstral Distance between the original and the quantized parameters
Trellis Coded Residual VQ

• Test sentences:
  • "Why were you away a year Roy ?" (Voiced);
  • "Nanny may know my meaning" (Nasals);
  • "The little blanket lies around on the floor" (Plosives);
  • "His vicious father has seizures" (Fricatives);
  • "The problem with swimming is that you can drown" (Voiced Fricatives);
  • "Which tea-party did Baker go to ?" (Plosives and Unvoiced Stops).
Trellis Coded Residual VQ
Trellis Coded Residual VQ

- Code vectors optimality conditions
- Experiments on random and natural sources
  - Sequential greedy codebook design
  - Viterbi and Exhaustive search
- Low memory and time complexity
- Good performance on low bit rates (comparable with ESVQ)
- Performance degradation when the number of stages increases
- Progressive encoding
Areas of research

• Lossy
  • Trellis Coded Vector Quantization
  • H.263+ frame skipping optimization
  • JPEG domain processing

• Lossless
  • **Linear Prediction and Classification**
  • Polynomial texture maps
  • Discrete colors images
Lossless Image Compression

- JPEG-LS call for contributions
- Low complexity, effective algorithms (LOCO-I, CALIC, UCM, Sunset, ...)
- Very hard to improve compression:
  - Global optimization ineffective
  - Linear prediction inappropriate
  - CALIC close to image entropy
- TMW (1997) improves upon CALIC
Adaptive Linear Prediction and Classification

- Single-step lossless coding algorithm proposed by us in 1999
- Combines adaptive linear predictors and classification
- Predictors are optimized pixel-by-pixel
- Prediction error is entropy coded
- Exploits local image statistics
Adaptive Linear Prediction and Classification

Explicit use of local statistics to:
- Classify the context of the current pixel
- Find the best Linear Predictor
Adaptive Linear Prediction and Classification

- Causal pixels with Manhattan distance of d or less (d=2)
- Fixed shape
- Weights \( w_0, \ldots, w_{i-1} \) optimized to minimize error energy inside \( W_{x,y}(R_p) \)

**Prediction:** \( I'(x,y) = \text{int}(w_0 \cdot I(x,y-2) + w_1 \cdot I(x-1,y-1) + w_2 \cdot I(x,y-1) + w_3 \cdot I(x+1,y-1) + w_4 \cdot I(x-2,y) + w_5 \cdot I(x-1,y)) \)

**Error:** \( \text{Err}(x,y) = I'(x,y) - I(x,y) \)
Adaptive Linear Prediction and Classification

Statistics collected inside the window $W_{x,y}(R_p)$

Not all samples in $W_{x,y}(R_p)$ are used to refine the predictor
Adaptive Linear Prediction and Classification

for every pixel $I(x,y)$ do begin

/* Classification */
Collect samples in $W_{x,y}(R_p)$
Select samples with context closer to the context of the current pixel $I(x,y)$

/* Prediction */
Compute a predictor $P$ from new samples
Encode and send the prediction error $ERR(x,y)$

end
Adaptive Linear Prediction and Classification

Standard “pgm” images, 256 greylevels (8 bits)

Balloon Barb Barb2 Board Boats
Girl Gold Hotel Zelda
Adaptive Linear Prediction and Classification

Experiments

• Predictor computation
  • Gradient Descend
  • Least Square Minimization

• Classification
  • LBG
  • Minimum Distance

• Parameters (window radius, # of predictors, context size,…)

• Different entropy coders
Adaptive Linear Prediction and Classification

Gradient Descend and LBG Classification
Compression rate in bits per pixel.
(# of predictors = 2, $R_p=10$)

<table>
<thead>
<tr>
<th></th>
<th>balloon</th>
<th>barb</th>
<th>barb2</th>
<th>board</th>
<th>boats</th>
<th>girl</th>
<th>gold</th>
<th>hotel</th>
<th>zelda</th>
<th>Avg.</th>
</tr>
</thead>
<tbody>
<tr>
<td>SUNSET</td>
<td>2.89</td>
<td>4.64</td>
<td>4.71</td>
<td>3.72</td>
<td>3.99</td>
<td>3.90</td>
<td>4.60</td>
<td>4.48</td>
<td>3.79</td>
<td>4.08</td>
</tr>
<tr>
<td>LOCO-I</td>
<td>2.90</td>
<td>4.65</td>
<td>4.66</td>
<td>3.64</td>
<td>3.92</td>
<td>3.90</td>
<td>4.47</td>
<td>4.35</td>
<td>3.87</td>
<td>4.04</td>
</tr>
<tr>
<td>UCM</td>
<td>2.81</td>
<td>4.44</td>
<td>4.57</td>
<td>3.57</td>
<td>3.85</td>
<td>3.81</td>
<td>4.45</td>
<td>4.28</td>
<td>3.80</td>
<td>3.95</td>
</tr>
<tr>
<td>Our</td>
<td>2.84</td>
<td>4.16</td>
<td>4.48</td>
<td>3.59</td>
<td>3.89</td>
<td>3.80</td>
<td>4.42</td>
<td>4.41</td>
<td>3.64</td>
<td>3.91</td>
</tr>
<tr>
<td>CALIC</td>
<td>2.78</td>
<td>4.31</td>
<td>4.46</td>
<td>3.51</td>
<td>3.78</td>
<td>3.72</td>
<td>4.35</td>
<td>4.18</td>
<td>3.69</td>
<td>3.86</td>
</tr>
<tr>
<td>TMW</td>
<td>2.65</td>
<td>4.08</td>
<td>4.38</td>
<td>3.27</td>
<td>3.61</td>
<td>3.47</td>
<td>4.28</td>
<td>4.01</td>
<td>3.50</td>
<td>3.70</td>
</tr>
</tbody>
</table>
Adaptive Linear Prediction and Classification

Least Squares Minimization
Solution of the system of linear equations:
\[ A_{x,y} = w_{x,y} * b_{x,y} \]
Where:
\[ A_i = c_i * c_i^T \]
\[ b_i = p_i * c_i \]
\[ A_{x,y} = \sum_i A_i \]
\[ b_{x,y} = \sum_i b_i \]
The column vector \( c_i \) is the context of the pixel \( p_i \).
Adaptive Linear Prediction and Classification

Entropy Coding

- Laplacian distribution
- Golomb (with mapping)
- Arithmetic coding
  - Zero Probability Problem

\[ p_i = \frac{(c_i + 1)}{(\sum_i c_i + n)} \]
Adaptive Linear Prediction and Classification

![Bar graph showing comparison of different datasets with various methods: TMW, CALIC, ALPC - Gradient Descend, and ALPC - Least Squares. The x-axis represents different datasets like Ballon, Barb2, Barb, Board, Boats, Girl, Gold, Hotel, Zelda, and TOTAL, while the y-axis represents the values from 2 to 4.5. The graph compares the performance of different methods across these datasets.](image-url)
Adaptive Linear Prediction and Classification

Test image “Board” and magnitude of the prediction error
Adaptive Linear Prediction and Classification

Test image “Hotel” and magnitude of the prediction error
Adaptive Linear Prediction and Classification

• Good compression when structures and textures are present

• Poor compression in high contrast zones

• Local Adaptive LP captures features not exploited by other systems
Areas of research

• Lossy
  • Trellis Coded Vector Quantization
  • H.263+ frame skipping optimization
  • JPEG domain processing

• Lossless
  • Linear Prediction and Classification
  • Polynomial texture maps
  • Discrete colors images
Variability in video sequences may cause the encoder to skip frames

In constant bit-rate encoding, frame skipping occurs frequently after a “scene change”

Assumption: encoder has some look-ahead capability
Low Bit-Rate Video Coding

H.263+
• State of the art Video Coding (MPEG-4 core)
• MC-prediction and DCT coding
• I and P macroblocks
• Frame and MB layer rate control heuristics
H.263+ Frame Skipping Optimization

Frame Layer Bit Rate Control

• Encoder monitors the transmission buffer and skips frames while the buffer holds more than $M$ bytes
• Buffer content is never negative (causal)
H.263+ Frame Skipping Optimization

Bits per frame (sequence “Std100.qcif”)
H.263+ Frame Skipping Optimization

PSNR and Bits per frame across a scene cut
H.263+ Frame Skipping Optimization

Optimal Strategy
Selects which frames should be encoded in order to minimize the number of skipped frames

Conditions:
- Skipping strategy is not changed
- Causal
H.263+ Frame Skipping Optimization

- Optimal algorithm proposed by us
- Algorithm based on dynamic programming
- Minimizes number of skipped frames
- PSNR and bit rate are not changed
- Improves H.263+ encoding
- Full compatibility with standard decoders
H.263+ Frame Skipping Optimization

• Minimizes the number of skipped frames while keeping the quality and bit rate constant

• Assumption:
  • When the quality of $F[i-d]$ is fixed, the cost $C[i,d]$ of predicting $F[i]$ from $F[i-d]$, does not depend on how $F[i-d]$ is encoded
**H.263+ Frame Skipping Optimization**

- \( F[i] = \) i-th frame in the sequence
- \( C[i, d] = \) cost (in bits) of predicting \( F[i] \) from \( F[i-d] \)  
  (If \( d=0, C[i, 0] = \) cost of \( F[i] \) I-coded)
- \( D = \) maximum number of frames that the encoder can skip (a constant that depends on the target bit rate)
- \( M = \) target bits per frame
H.263+ Frame Skipping Optimization

- $T[i, j] = \text{Number of transmitted frames}$
- $B[i, j] = \text{Corresponding buffer content}$
- $P[i, j] = \text{Row pointer to build the solution}$
- $d[i] = \text{Solution vector.}$

\[ d[i] = 0 \text{ if frame } F[i] \text{ is skipped} \]

Time complexity: $O(D^2 n) = O(n)$ (constant $D \approx 7$)
## H.263+ Frame Skipping Optimization

### Cost Matrix $C[i][j]$:

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>32</th>
<th>28</th>
<th>43</th>
<th>38</th>
<th>31</th>
<th>35</th>
<th>40</th>
<th>41</th>
<th>45</th>
<th>44</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>32</td>
<td>30</td>
<td>25</td>
<td>33</td>
<td>34</td>
<td>27</td>
<td>29</td>
<td>34</td>
<td>32</td>
<td>40</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>24</td>
<td>30</td>
<td>41</td>
<td>24</td>
<td>32</td>
<td>28</td>
<td>21</td>
<td>25</td>
<td>31</td>
<td>28</td>
<td>34</td>
<td>35</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>23</td>
<td>17</td>
<td>23</td>
<td>32</td>
<td>19</td>
<td>28</td>
<td>23</td>
<td>18</td>
<td>21</td>
<td>26</td>
<td>24</td>
<td>28</td>
<td>31</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>22</td>
<td>19</td>
<td>21</td>
<td>18</td>
<td>20</td>
<td>15</td>
<td>22</td>
<td>18</td>
<td>13</td>
<td>18</td>
<td>20</td>
<td>16</td>
<td>16</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>30</td>
<td>30</td>
<td>44</td>
<td>25</td>
<td>15</td>
<td>30</td>
<td>50</td>
<td>29</td>
<td>36</td>
<td>58</td>
<td>50</td>
<td>40</td>
<td>34</td>
<td>27</td>
<td></td>
</tr>
</tbody>
</table>

### Buffer Matrix $B[i][j]$:

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>32</th>
<th>28</th>
<th>43</th>
<th>38</th>
<th>31</th>
<th>35</th>
<th>40</th>
<th>41</th>
<th>45</th>
<th>44</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>22</td>
<td>20</td>
<td>15</td>
<td>23</td>
<td>24</td>
<td>17</td>
<td>19</td>
<td>24</td>
<td>22</td>
<td>30</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>20</td>
<td>31</td>
<td>14</td>
<td>22</td>
<td>20</td>
<td>11</td>
<td>17</td>
<td>21</td>
<td>18</td>
<td>24</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>7</td>
<td>13</td>
<td>22</td>
<td>14</td>
<td>18</td>
<td>17</td>
<td>15</td>
<td>18</td>
<td>17</td>
<td>22</td>
<td>25</td>
<td>29</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>11</td>
<td>15</td>
<td>10</td>
<td>5</td>
<td>17</td>
<td>8</td>
<td>11</td>
<td>8</td>
<td>18</td>
<td>6</td>
<td>12</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>10</td>
<td>0</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

### Number of Transmitted Frames $T[i][j]$:

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>2</th>
<th>2</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>4</th>
<th>5</th>
<th>5</th>
<th>5</th>
<th>6</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>6</td>
<td>6</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>5</td>
<td>5</td>
<td>6</td>
<td>6</td>
<td>7</td>
<td>7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>5</td>
<td>4</td>
<td>6</td>
<td>6</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Pointers Matrix $P[i][j]$:

<table>
<thead>
<tr>
<th></th>
<th>-1</th>
<th>-1</th>
<th>-1</th>
<th>-1</th>
<th>-1</th>
<th>-1</th>
<th>-1</th>
<th>-1</th>
<th>-1</th>
<th>-1</th>
<th>-1</th>
<th>-1</th>
<th>-1</th>
<th>-1</th>
<th>-1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

### Decision Vector $D[i]$:

| 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |
H.263+ Frame Skipping Optimization

Std and Std100 - Concatenation of standard sequences

Commercials - continuous sampling of TV commercials
H.263+ Frame Skipping Optimization

![Bar Chart](chart.png)
### H.263+ Frame Skipping Optimization

#### Bit Rate and PSNR

<table>
<thead>
<tr>
<th>Bit Rate</th>
<th>TMN-8</th>
<th>Optimal</th>
</tr>
</thead>
<tbody>
<tr>
<td>commercials</td>
<td>32.31</td>
<td>32.22</td>
</tr>
<tr>
<td>commercials1</td>
<td>32.27</td>
<td>32.15</td>
</tr>
<tr>
<td>commercials2</td>
<td>32.55</td>
<td>32.54</td>
</tr>
<tr>
<td>commercials3</td>
<td>32.13</td>
<td>32.01</td>
</tr>
<tr>
<td>commercials4</td>
<td>32.44</td>
<td>32.11</td>
</tr>
<tr>
<td>commercials5</td>
<td>32.77</td>
<td>32.59</td>
</tr>
<tr>
<td>std</td>
<td>32.54</td>
<td>32.55</td>
</tr>
<tr>
<td>std100</td>
<td>32.57</td>
<td>32.60</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>PSNR_Y</th>
<th>TMN-8</th>
<th>Optimal</th>
</tr>
</thead>
<tbody>
<tr>
<td>commercials</td>
<td>27.97</td>
<td>28.00</td>
</tr>
<tr>
<td>commercials1</td>
<td>27.04</td>
<td>27.05</td>
</tr>
<tr>
<td>commercials2</td>
<td>28.83</td>
<td>28.84</td>
</tr>
<tr>
<td>commercials3</td>
<td>29.06</td>
<td>29.07</td>
</tr>
<tr>
<td>commercials4</td>
<td>28.46</td>
<td>28.54</td>
</tr>
<tr>
<td>commercials5</td>
<td>27.72</td>
<td>27.71</td>
</tr>
<tr>
<td>std</td>
<td>32.60</td>
<td>32.61</td>
</tr>
<tr>
<td>std100</td>
<td>31.74</td>
<td>31.76</td>
</tr>
</tbody>
</table>
H.263+ Frame Skipping Optimization

TMN-8

Heuristic
### H.263+ Frame Skipping Optimization

**Encode**

<table>
<thead>
<tr>
<th>Frame No.</th>
<th>n-5</th>
<th>n-4</th>
<th>n-3</th>
<th>n-2</th>
<th>n-1</th>
<th>n</th>
<th>n+1</th>
<th>n+2</th>
<th>n+3</th>
<th>n+4</th>
<th>n+5</th>
<th>n+6</th>
<th>n+7</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Encode</td>
<td>n-5</td>
<td>n-4</td>
<td>n-3</td>
<td>n-2</td>
<td>n-1</td>
<td>n</td>
<td>skip</td>
<td>n+4</td>
<td>skip</td>
<td>n+6</td>
<td>n+7</td>
<td>...</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Transmit**

<table>
<thead>
<tr>
<th>Frame No.</th>
<th>n-5</th>
<th>n-4</th>
<th>n-3</th>
<th>n-2</th>
<th>n-1</th>
<th>n</th>
<th>n+4</th>
<th>n+6</th>
<th>n+7</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transmit</td>
<td>n-5</td>
<td>n-4</td>
<td>n-3</td>
<td>n-2</td>
<td>n-1</td>
<td>n</td>
<td>n+4</td>
<td>n+6</td>
<td>n+7</td>
<td>...</td>
</tr>
</tbody>
</table>

**Display**

<table>
<thead>
<tr>
<th>Frame No.</th>
<th>n-5</th>
<th>n-4</th>
<th>n-3</th>
<th>n-2</th>
<th>n-1</th>
<th>n</th>
<th>n+4</th>
<th>n+6</th>
<th>n+7</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display</td>
<td>...</td>
<td>n-5</td>
<td>n-4</td>
<td>n-3</td>
<td>n-2</td>
<td>n-1</td>
<td>n</td>
<td>n+4</td>
<td>n+6</td>
<td>n+7</td>
</tr>
</tbody>
</table>

**TMN-8**

Heuristic - Encode last frame of the skipped sequence
H.263+ Frame Skipping Optimization

<table>
<thead>
<tr>
<th>Skipped</th>
<th>TMN-8</th>
<th>Optimal</th>
<th>Heuristic</th>
</tr>
</thead>
<tbody>
<tr>
<td>commercials</td>
<td>1782</td>
<td>1597</td>
<td>1605</td>
</tr>
<tr>
<td>commercials1</td>
<td>2151</td>
<td>1858</td>
<td>1866</td>
</tr>
<tr>
<td>commercials2</td>
<td>1063</td>
<td>940</td>
<td>947</td>
</tr>
<tr>
<td>commercials3</td>
<td>1099</td>
<td>936</td>
<td>985</td>
</tr>
<tr>
<td>commercials4</td>
<td>1897</td>
<td>1697</td>
<td>1700</td>
</tr>
<tr>
<td>commercials5</td>
<td>1386</td>
<td>1275</td>
<td>1282</td>
</tr>
<tr>
<td>std</td>
<td>204</td>
<td>177</td>
<td>196</td>
</tr>
<tr>
<td>std100</td>
<td>396</td>
<td>343</td>
<td>376</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>PSNR_Y</th>
<th>TMN-8</th>
<th>Optimal</th>
<th>Heuristic</th>
</tr>
</thead>
<tbody>
<tr>
<td>commercials</td>
<td>27.97</td>
<td>28.00</td>
<td>27.99</td>
</tr>
<tr>
<td>commercials1</td>
<td>27.04</td>
<td>27.05</td>
<td>27.05</td>
</tr>
<tr>
<td>commercials2</td>
<td>28.83</td>
<td>28.84</td>
<td>28.84</td>
</tr>
<tr>
<td>commercials3</td>
<td>29.06</td>
<td>29.07</td>
<td>29.09</td>
</tr>
<tr>
<td>commercials4</td>
<td>28.46</td>
<td>28.54</td>
<td>28.52</td>
</tr>
<tr>
<td>commercials5</td>
<td>27.72</td>
<td>27.71</td>
<td>27.70</td>
</tr>
<tr>
<td>std</td>
<td>32.60</td>
<td>32.61</td>
<td>32.60</td>
</tr>
<tr>
<td>std100</td>
<td>31.74</td>
<td>31.76</td>
<td>31.75</td>
</tr>
</tbody>
</table>
H.263+ Frame Skipping Optimization

Unrestricted optimization

• Causality constraint is removed

• NP-Complete
  • Formulation of the corresponding decision problem
  • Proof by reduction to LONGEST PATH
H.263+ Frame Skipping Optimization

Reduction to LONGEST PATH

- Input: a graph $G(V, E)$
- $n = |V|$
- $M = 1$
- $C[i, j] = 1$ if $(v_i, v_j)$ is in $E$
- $0$ otherwise
H.263+ Frame Skipping Optimization

• Optimization substantially reduces frame skipping
• Effective method to improve quality in proximity of scene cuts. Bit rate is not increased
• Simple heuristic gets results close to optimal solution
• Suitable for encoders of the MPEG family, provided that encoder has look-ahead capability
• Decoding is unaffected
• Unrestricted optimization is NP-complete
Conclusions

• Trellis Coded Vector Quantization
  • New quantizer proposed and studied
  • Optimal conditions derived
  • Experiments on random sources, images, speech coding

• Linear Prediction and classification
  • New algorithm for lossless image compression
  • Improves upon single pass, state of the art

• H.263+ frame skipping optimization
  • New Frame layer rate control
  • Optimal dynamic programming algorithm
  • Effective heuristic inspired by the optimal algorithm
  • NP-completeness of the unrestricted problem