Implementation of an Unequal Error Protection Scheme for Scalable Foveated Image Communication

Embedded Software Systems Course Project
May 1, 2002

Muhammad Farooq Sabir
Rashmi Tripathi
Problem Statement

- To provide Unequal Error Protection (UEP) against the channel noise, for real-time scalable image communication.
- **Embedded Foveation Image Coding (EFIC)**
  - Bits with greater contribution to the foveated visual distortion are encoded and transmitted first.
- **Scalable Image Communication**
  - Bitstream can be truncated at any point to provide different compression ratios.
- **Tradeoff**
  - Decrease protection as the importance of bits decreases, in order to obtain high raw data transmission rate.
UEP using Punctured Turbo codes

Use rate compatible punctured turbo codes to provide different level of error protection to different portions of the bitstream.
Implementation

- Each block in the system is modeled as an SDF actor.
- **Punctured Turbo Encoder:**
  - Rate 1/3 encoder with 16 states.
  - 8 different levels of puncturing for different portions of the bitstream.
  - Fixed point implementation.
- **Turbo Decoder:**
  - Uses Soft Output Viterbi Algorithm (SOVA).
    - 3 times lower complexity as compared to Maximum A posteriori Probability (MAP) algorithm.
  - Allows low complexity decoding.
  - Floating point implementation.
Implementation contd.

- Comparing MAP and SOVA performance

- Modules have been written in C.
- System is being implemented on TMS320C6701 floating-point DSP.
Results

- Encoder has been optimized.
- Decoder has been optimized with respect to memory.
- **Optimization Statistics:**

<table>
<thead>
<tr>
<th>Encoder</th>
<th>Decoder</th>
</tr>
</thead>
<tbody>
<tr>
<td>Optimization Stage</td>
<td>No. of Instruction Cycles (in Millions)</td>
</tr>
<tr>
<td>Without any optimization</td>
<td>11.18</td>
</tr>
<tr>
<td>After Level 3 optimization</td>
<td>5.23</td>
</tr>
<tr>
<td>After memory optimization (code &amp; data)</td>
<td>1.81</td>
</tr>
<tr>
<td>After loop unrolling</td>
<td>1.09</td>
</tr>
<tr>
<td>After coding in assembly</td>
<td>0.48</td>
</tr>
</tbody>
</table>
Results contd.

- Comparison between Uniform (rate 2/3) and Unequal (overall rate 3/4) Error Protection.

![BER vs. SNR for EFIC at compression ratio of 8:1](image1)
![BER vs. SNR for EFIC, truncated to give a compression ratio of 32:1](image2)
Conclusions

- An unequal error protection scheme for EFIC compressed images using Punctured Turbo Codes
  - Written in C and implemented on TMS320C6701 DSP processor.
  - Optimized with respect to memory and computation time.

- Presently working on
  - Optimization of ‘Puncture’ and ‘Insert Zeros’ blocks.
  - Assembly level optimization of the decoder.

- Future Work
  - Fixed point implementation of the decoder.
  - Implementation of unequal error protection employing spatial diversity, as a real-time embedded system.