NETVC Hackathon Results
IETF 93 (Prague)
• Integrated Thor into *AreWeCompressedYet*
  
  - Had to disable B-frames
    
    - Thor required frame count to be a multiple of the GOP size (12 frames)
Thor and Daala: PSNR

Rate: -43.54906%   DSNR: 1.72357 dB
Thor and Daala: PSNR

Rate: -23.64632%  DSNR: 0.92675 dB
Thor and Daala: PSNR-HVS

Rate: -0.79660%  
DSNR: 0.22439 dB
Thor and Daala: MS FastSSIM

Rate: 91.02910%  DSNR: -1.59731 dB
Daala with Thor MC

• Daala’s motion compensation is very loosely coupled with the rest of the codec
• Ripped it out and replaced it with Thor’s
  – Ran whole Thor encoder with residual coding disabled, stuffed output bits into a Daala frame
• Performed four different variations of this experiment
Daala Inter Encoder

- Input Frame
  - Reference Frames
    - OBMC
  - Prediction Frame
    - Forward Transform
    - Prediction Coefficients
    - Forward Transform
    - Prediction Coefficients
      - DCs and PVQ data
    - PVQ
      - Entropy Coding
      - Bitstream
        - Bitstream
          - Dequant + Inverse Transform
            - Decoded Image
              - Input Frame
                - Input Coefficients
                  - Forward Transform
Daala Inter Encoder

- Input Frame
- Reference Frames
- Prediction Frame
- Decoded Image
- DCs and PVQ data
- Bitstream
- Forward Transform
- Prediction Coefficients
- Entropy Coding
- PVQ
- Dequant + Inverse Transform
- Thor
- OBMC
- DCs and PVQ data
- DCs and PVQ data
- Thor
Experiment #1

- Disabled residual coding
- Disabled intra modes
- Disabled 64x64 blocks

<table>
<thead>
<tr>
<th>Metric</th>
<th>derf-exp-thor-mc1b</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSNR</td>
<td>24.51603</td>
</tr>
<tr>
<td>PSNRHVS</td>
<td>24.33240</td>
</tr>
<tr>
<td>SSIM</td>
<td>27.75545</td>
</tr>
<tr>
<td>FASTSSIM</td>
<td>24.36346</td>
</tr>
</tbody>
</table>

master-2015-07-17-37fa007 → derf-exp-thor-mc1b
Experiment #2

- Stop coding CBP bits
- Omit disabled modes from VLCs
- Omit 64x64 → 32x32 split bits

```
master-2015-07-17-37fa007 → derf-exp-thor-mc2b

<table>
<thead>
<tr>
<th>Metric</th>
<th>PSNR</th>
<th>DSNR (dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSNR</td>
<td>7.93727</td>
<td>-0.25080</td>
</tr>
<tr>
<td>PSNRHVS</td>
<td>7.13015</td>
<td>-0.35463</td>
</tr>
<tr>
<td>SSIM</td>
<td>9.18280</td>
<td>-0.22162</td>
</tr>
<tr>
<td>FASTSSIM</td>
<td>7.28931</td>
<td>-0.20654</td>
</tr>
</tbody>
</table>
```
Experiment #3

- Re-enable intra modes
- Add them back to the VLCs
Experiment #4

- Re-enable 64x64 blocks
- Code 64x64 → 32x32 splits again

```
derf-exp-thor-mc+intra → derf-exp-thor-mc+intra+64x64
```

<table>
<thead>
<tr>
<th>Metric</th>
<th>Rate (%)</th>
<th>DSNR (dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSNR</td>
<td>-11.75578</td>
<td>0.35987</td>
</tr>
<tr>
<td>PSNRHVS</td>
<td>-9.23958</td>
<td>0.43580</td>
</tr>
<tr>
<td>SSIM</td>
<td>-10.83236</td>
<td>0.25502</td>
</tr>
<tr>
<td>FASTSSIM</td>
<td>-7.25009</td>
<td>0.17587</td>
</tr>
</tbody>
</table>
Experiment #4 vs. Daala

- Re-enable 64x64 blocks
- Code 64x64 → 32x32 splits again

<table>
<thead>
<tr>
<th>Metric</th>
<th>PSNR</th>
<th>DSNR</th>
</tr>
</thead>
<tbody>
<tr>
<td>master-2015-07-17-37fa007 → derf-exp-thor-mc+intra+64x64</td>
<td>-6.45022</td>
<td>0.18862</td>
</tr>
<tr>
<td>PSNRHVS</td>
<td>-4.37206</td>
<td>0.19331</td>
</tr>
<tr>
<td>SSIM</td>
<td>-4.55062</td>
<td>0.09759</td>
</tr>
<tr>
<td>FASTSSIM</td>
<td>0.52045</td>
<td>-0.03280</td>
</tr>
</tbody>
</table>

![Graph showing relationship between bits per pixel and picture quality (dB)]
Thor’s Constrained Low-Pass Filter

• Easy to integrate into Daala
  – Simple hack with no signaling for now (e.g., cheating)
  – Better patch with real signaling in progress

• Solves a long-standing “quilting” artifact
  – Found in fades at low rates
Previous Daala (no CLPF)

Fade from Sintel at -v70 (brightness and contrast enhanced)
Daala with Thor’s CLPF

Fade from Sintel at -v70 (brightness and contrast enhanced)