Thor update

High Efficiency, Moderate Complexity Video Codec using only RF IPR
(https://datatracker.ietf.org/ipr/2636/)

draft-fuldseth-netvc-thor-03
Steinar Midtskogen (Cisco)
IETF 98 – Chicago, US – March 2017
Changes in Thor since IETF97

• Support for monochrome video added
• Support for video with 4:2:2 chroma sampling
  – encoded as 4:4:4 internally
    • very simple to implement and the code remains clean and simple
    • suboptimal compression, 4:2:2 a corner case, complexity avoided for something few will use
• The constrained low-pass filter (CLPF) improved
  – Gives ~0.4% BDR gain
• Misc fixes (hbd CFL, portability, code sync with AV1)
Improved CLPF

- Increase from 6 to 8 taps:

- A ramp-down added to the clip function
  - Most of the gain comes from this new clip function
Improved CLPF

• New clip function:

  ```c
  int constrain(int x, int s, unsigned int bitdepth) {
    return (x < 0 ? -1 : 1) * 
      max(0, abs(x) - max(0, abs(x) - s + (abs(x) >> (bitdepth - 3 - log2(s)))));
  }

  int clpf_pixel(int X, int A, int B, int C, int D, int E, int F, int G, int H, 
                 int s, unsigned int bitdepth) {
    int delta = 1 * constrain(A - X, s, bitdepth) + 3 * constrain(B - X, s, bitdepth) + 
                1 * constrain(C - X, s, bitdepth) + 3 * constrain(D - X, s, bitdepth) + 
                3 * constrain(E - X, s, bitdepth) + 1 * constrain(F - X, s, bitdepth) + 
                3 * constrain(G - X, s, bitdepth) + 1 * constrain(H - X, s, bitdepth); 
    return X + (8 + delta - (delta < 0)) >> 4;  // Rounding, assumes arithmetic shift
  }
  ```

• Simple C code, SIMD friendly and easy to implement:
Thor compared to AV1

- Much activity in AV1 recently, not so much in Thor
- Codec performance now measured using AWCY
- AV1 now generally performs better than Thor
  - Many AV1 improvements and new tools
  - Thor may still be slightly better at low delay videoconferencing (meeting rooms and "talking heads")
  - AV1 is much better than Thor at screen content
  - AV1 seems to have a 3-4x speed advantage over Thor on the AWCY servers compared to our servers, unknown reason
  - Error resilience has a significant cost
- AV1 is a moving target and will probably improve further by ~10% in the coming months
Low delay, error resilient
Low delay, not error resilient
High delay, error resilient
High delay, not error resilient
Differing results

- The results depend a lot on the sequence test
- Change from Thor to AV1 at similar complexity for some sequences as reported by AWCY:

<table>
<thead>
<tr>
<th>Sequence</th>
<th>20.18</th>
<th>18.77</th>
<th>36.50</th>
<th>18.26</th>
<th>13.64</th>
<th>19.14</th>
<th>19.99</th>
<th>11.86</th>
<th>18.85</th>
<th>24.84</th>
</tr>
</thead>
<tbody>
<tr>
<td>kirland360p_60f.y4m</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>life_1080p30_60f.y4m</td>
<td>-17.24</td>
<td>-19.94</td>
<td>-20.82</td>
<td>-22.77</td>
<td>-38.85</td>
<td>-38.43</td>
<td>-17.53</td>
<td>-38.56</td>
<td>-38.29</td>
<td>-20.16</td>
</tr>
<tr>
<td>niklas360p_60f.y4m</td>
<td>-3.93</td>
<td>-7.02</td>
<td>-1.43</td>
<td>-12.31</td>
<td>-22.83</td>
<td>-23.46</td>
<td>-4.07</td>
<td>-22.83</td>
<td>-23.75</td>
<td>-6.64</td>
</tr>
<tr>
<td>rush_hour_1080p25_60f.y4m</td>
<td>13.00</td>
<td>10.98</td>
<td>12.67</td>
<td>4.87</td>
<td>-8.47</td>
<td>-15.50</td>
<td>12.71</td>
<td>-8.81</td>
<td>-15.72</td>
<td>11.67</td>
</tr>
<tr>
<td>shields_640x360_60f.y4m</td>
<td>-10.65</td>
<td>-10.27</td>
<td>-8.30</td>
<td>-8.80</td>
<td>-16.18</td>
<td>-19.34</td>
<td>-9.71</td>
<td>-16.84</td>
<td>-18.84</td>
<td>-8.83</td>
</tr>
<tr>
<td>speed_bag_640x360_60f.y4m</td>
<td>-16.05</td>
<td>-11.34</td>
<td>-6.60</td>
<td>-22.36</td>
<td>-32.54</td>
<td>-30.55</td>
<td>-17.43</td>
<td>-33.44</td>
<td>-32.83</td>
<td>-10.09</td>
</tr>
<tr>
<td>thaloundeskmgt360p_60f.y4m</td>
<td>-43.41</td>
<td>-38.83</td>
<td>-38.29</td>
<td>-43.92</td>
<td>-49.93</td>
<td>-48.02</td>
<td>-38.10</td>
<td>-48.61</td>
<td>-47.31</td>
<td>-38.01</td>
</tr>
<tr>
<td>touchdown_pass_1080p_60f.y4m</td>
<td>-0.37</td>
<td>-4.57</td>
<td>-5.88</td>
<td>-2.04</td>
<td>-24.73</td>
<td>-16.97</td>
<td>-2.63</td>
<td>-25.82</td>
<td>-18.21</td>
<td>-4.85</td>
</tr>
<tr>
<td>vidyo1_720p_60fps_60f.y4m</td>
<td>4.62</td>
<td>0.94</td>
<td>12.39</td>
<td>-2.04</td>
<td>-15.03</td>
<td>-21.51</td>
<td>4.51</td>
<td>-15.57</td>
<td>-21.50</td>
<td>7.87</td>
</tr>
<tr>
<td>vidyo4_720p_60fps_60f.y4m</td>
<td>5.57</td>
<td>2.55</td>
<td>8.21</td>
<td>-5.66</td>
<td>-27.02</td>
<td>-21.97</td>
<td>5.09</td>
<td>-26.85</td>
<td>-21.72</td>
<td>5.96</td>
</tr>
<tr>
<td>wikipedia_420.y4m</td>
<td>-79.82</td>
<td>-79.88</td>
<td>-80.09</td>
<td>-79.69</td>
<td>-79.97</td>
<td>-80.64</td>
<td>-79.60</td>
<td>-79.84</td>
<td>-80.57</td>
<td>-80.22</td>
</tr>
</tbody>
</table>
The road ahead (thinking loudly)

- Add arithmetic coding
  - Use the Daala entropy coder?
  - The entropy coder is in the core of a codec, so this should perhaps rather be regarded as a merge of Thor and Daala

- Merge CLPF and Daala dering (AV1 CDEF)

- Since the above tools have been adopted in AV1, this path would take the codec towards (a subset of) AV1
  - Other tools from Thor already adopted in AV1:
    - 7 bit interpolation filters
    - quantisation matrices
    - delta-q