Effective Visual Masking Techniques in JPEG2000

Thomas Richter, University of Stuttgart

Rate allocation in the JPEG2000 image compression algorithm [1] is performed by the EBCOT algorithm [2], measures file size and distortion, defined as mean square error (MSE). Since MSE correlates only mediocre to visual quality, more advanced metrics like the M-SSIM [6] have been proposed. One exploitable effect of the human visual system is that of visual masking: If a structure of a fixed amplitude is overlayed by a texture, it becomes masked and less visible [4]. This can be addressed in JPEG2000 by multiplying the MSE contribution of a codeblock [1] by a factor $\mu$ computed from the neighbourhood of the data [2], [3]. Most of these techniques require, however, complex operations on the coefficients.

A low complexity technique is to measure the moments of the wavelet highpasses, fit their amplitude density to the GGD model [5] $p_m(y) \sim \exp(y/s)$ and compute $\mu$ indirectly from $p_m$. The same data can also be used to speed up JPEG2000 compression by a factor of two on average [7]. The suggested algorithm to compute $\mu$ is as follows:

- Measure the second moment $\sigma^2 = \mathbb{E}(X^2)$ and $\zeta = \mathbb{E}(|X|)$ per codeblock.
- Compute $\zeta^2/\sigma^2 = \Gamma(2/\alpha)^2/\Gamma(3/\alpha)\Gamma(1/\alpha)$ to find $\alpha$ and $s = \zeta \Gamma(1/\alpha)/\Gamma(2/\alpha)$.
- Compute the MSE weight $\mu := C_v C_c \left(\sqrt{g/s} \cdot \exp(-q/s) + \exp(-s/q)\right)$ where $g$ is the nominal gain, $C_v$ a frequency weight and $C_c$ a color weight.

This algorithm requires only two additions, one multiplication and one masking per coefficient. To measure the impact of visual masking, the author measured $20 \log(1 - \text{MSSIM})$ on an unmodified PSNR-optimal JPEG2000 implementation, a version extended by CSF weights following [1], and one using CSF weights and visual masking. For better locality, $32 \times 32$ codeblocks have been chosen.

<table>
<thead>
<tr>
<th>codec</th>
<th>bopp</th>
<th>regular</th>
<th>CSF</th>
<th>CSF+VM</th>
<th>bopp</th>
<th>regular</th>
<th>CSF</th>
<th>CSF+VM</th>
<th>bopp</th>
<th>regular</th>
<th>CSF</th>
<th>CSF+VM</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.8</td>
<td>30.6812</td>
<td>30.3646</td>
<td>30.6222</td>
<td>38.2613</td>
<td>40.8275</td>
<td>42.3146</td>
<td>31.9019</td>
<td>34.8310</td>
<td>38.0465</td>
<td>39.1321</td>
<td>40.6287</td>
<td>42.4918</td>
</tr>
<tr>
<td>1.0</td>
<td>31.2631</td>
<td>31.1551</td>
<td>36.3900</td>
<td>40.7098</td>
<td>45.6418</td>
<td>44.2968</td>
<td>35.3554</td>
<td>37.8566</td>
<td>38.7780</td>
<td>39.8555</td>
<td>41.0022</td>
<td>42.6648</td>
</tr>
<tr>
<td>1.4</td>
<td>37.1557</td>
<td>39.8555</td>
<td>41.0022</td>
<td>44.0197</td>
<td>44.6894</td>
<td>47.6454</td>
<td>39.3156</td>
<td>42.3944</td>
<td>42.8387</td>
<td>43.2419</td>
<td>46.0324</td>
<td>46.1150</td>
</tr>
<tr>
<td>1.6</td>
<td>39.1521</td>
<td>41.4555</td>
<td>44.9218</td>
<td>45.6884</td>
<td>48.1425</td>
<td>48.7465</td>
<td>41.9726</td>
<td>44.4015</td>
<td>44.8227</td>
<td>45.6684</td>
<td>48.1425</td>
<td>49.2908</td>
</tr>
<tr>
<td>1.8</td>
<td>40.3614</td>
<td>42.8149</td>
<td>44.9218</td>
<td>45.5724</td>
<td>49.2890</td>
<td>49.6699</td>
<td>43.4219</td>
<td>46.0324</td>
<td>46.1150</td>
<td>48.1425</td>
<td>49.2908</td>
<td>50.3455</td>
</tr>
</tbody>
</table>

As seen, introduction of CSF weights always improves the M-SSIM performance typically by about 2dB. Visual masking can also help noticeably and increases the quality by another 2dB.

REFERENCES