# Comparison of Code-Pass-Skipping Strategies for Accelerating a JPEG 2000 Decoder

Volker Bruns



Fraunhofer IIS

# Motivation

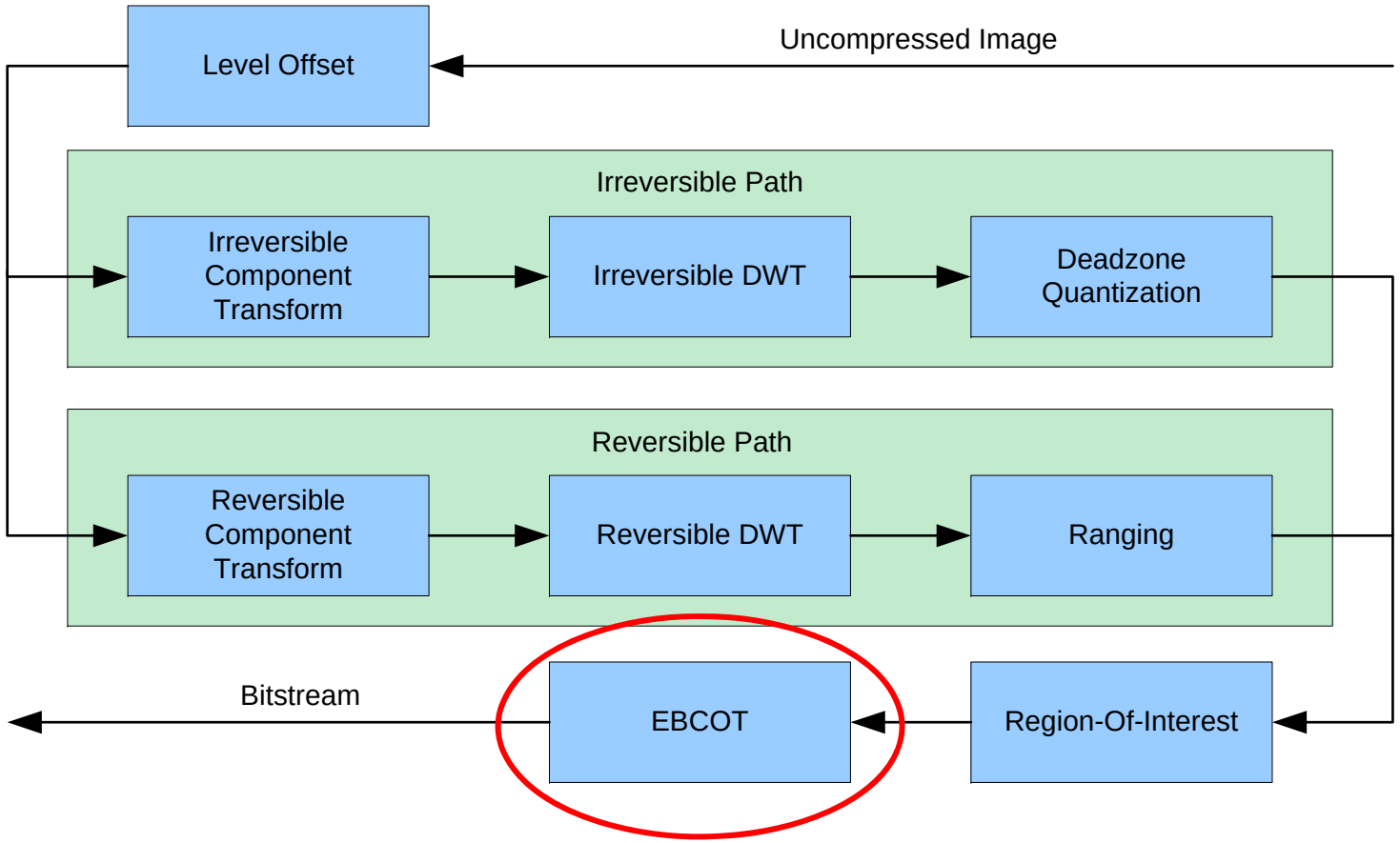
- Digital Cinema Packages (DCP)
  - data-rate up to 250 Mbit/s
  - 24 fps (per Eye)
  - 12 bit per color component
- Real-time processing very computationally expensive
  - Software-Codec (hexa-core CPU)
  - GPU-based codec (additional GPU with many CUDA cores)
- Cinema Servers have dedicated hardware
- Many scenarios where software solution is desired
  - Quality Control, Screenings, uncertified cinema servers, …

# Motivation (cont.)

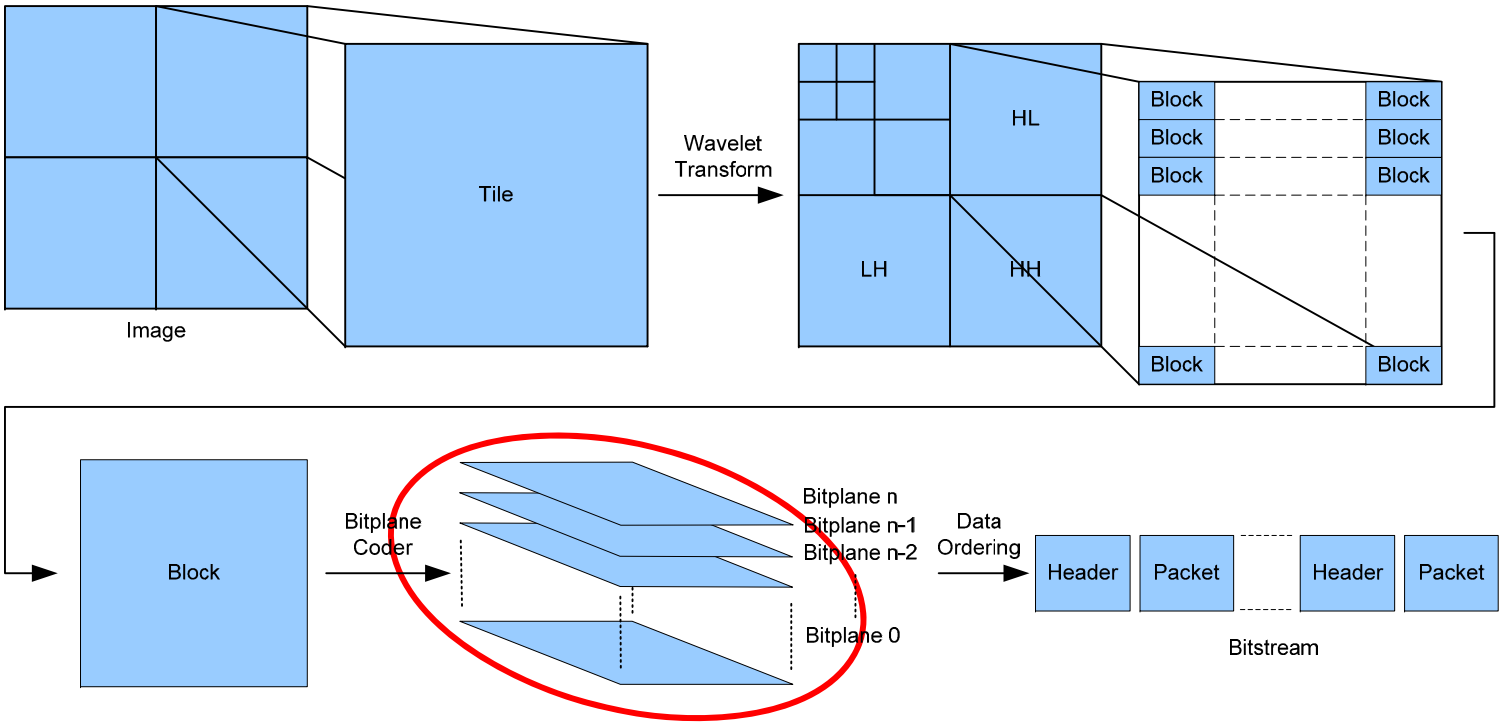
- Not all computers have sufficient computation power
  - Q: Utilize JPEG 2000's scalability by resolution and decode only 1K?
  - A: Displays are capable of displaying 2K, so not the smartest approach!
- Often, DCP is displayed with only 24-bit colors (on LCD, Consumer-Level Beamer)
  - Q: Why decode 36-bit in this case?
  - A: Better use Code-Pass-Skipping to sacrifice some precision and gain speed!

- Research Question:
  - How should a decoder decide which code-passes to ignore so as to maximize speed gain and minimize quality loss?
  - What is a good trade-off between speed gain and quality loss for the 24 DCP-preview scenario?

# JPEG 2000 Encoder



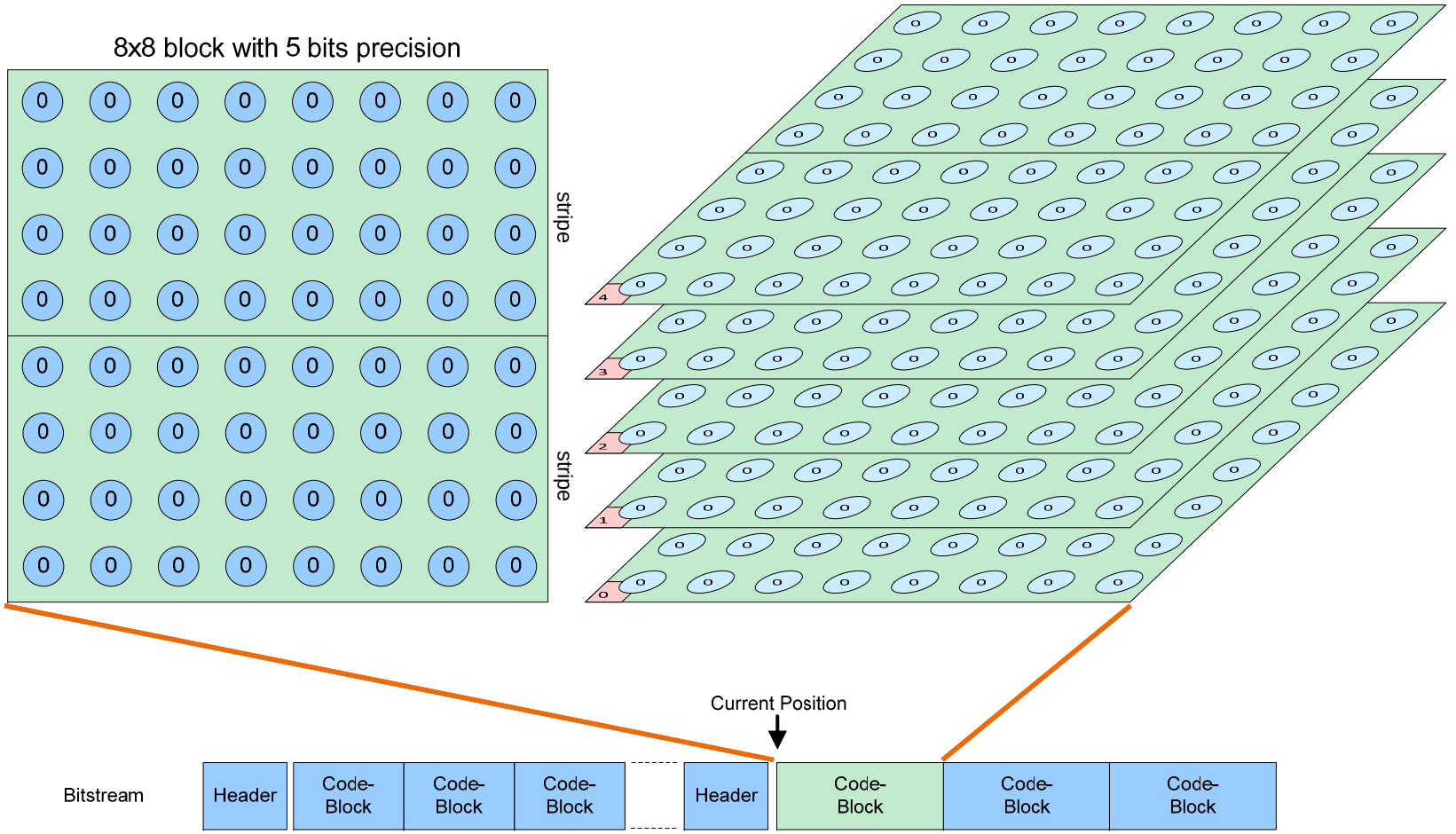
# JPEG 2000 Encoder (cont.)






Fraunhofer IIS

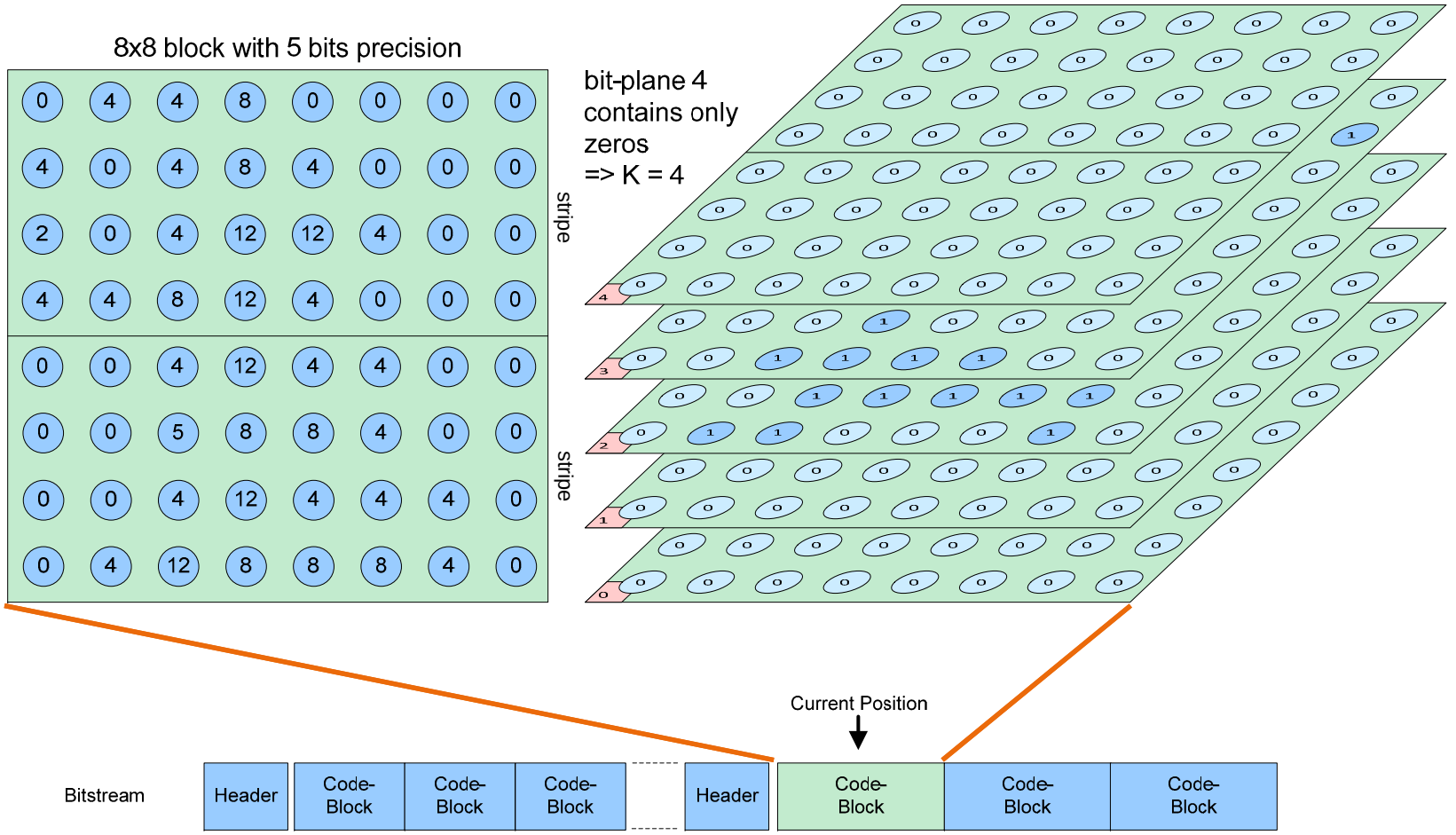
# Code-Block of Wavelet Coefficients






Fraunhofer IIS

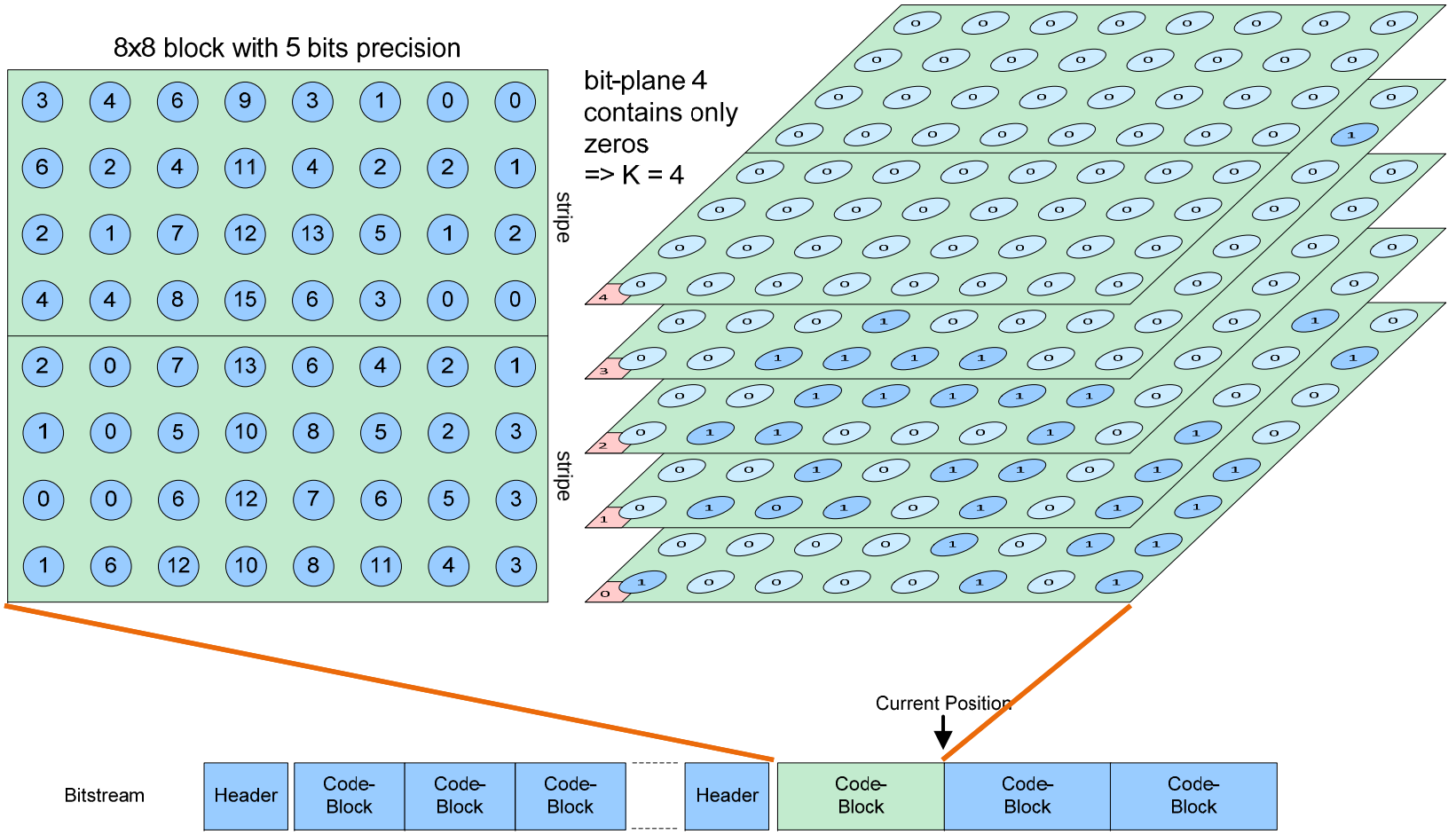
# Code-Block of Wavelet Coefficients






Fraunhofer IIS

# Code-Block of Wavelet Coefficients






Fraunhofer IIS

# Code-Pass-Histogram

- same image compressed at 75 Mbit/s and 250 Mbit/s
  - 75 Mbit/s: Encoder has already dropped some passes
  - 250 Mbit/s: Encoder has included all passes

Image 00777, 75 Mbit/s

| Bit-plane | Code-Pass | Total | rlvl 0 | rlvl 1 | rlvl 2 | rlvl 3 | rlvl 4 | rlvl 5 |
|---|---|---|---|---|---|---|---|---|
| Σ | Σ | 20792 | 164 | 379 | 1268 | 3971 | 8495 | 6515 |
| [...] | [...] | [...] | [...] | [...] | [...] | [...] | [...] | [...] |
| 9 | *27* | 1001 | 6 | 16 | 55 | 232 | 541 | 151 |
| | *26* | 715 | 6 | 16 | 55 | 232 | 401 | 5 |
| | *25* | 708 | 6 | 16 | 69 | 280 | 337 | 0 |
| 8 | 24 | 707 | 6 | 16 | 69 | 280 | 336 | 0 |
| | 23 | 516 | 6 | 16 | 69 | 280 | 145 | 0 |
| | 22 | 521 | 6 | 18 | 72 | 288 | 137 | 0 |
| 7 | 21 | 384 | 6 | 18 | 72 | 288 | 0 | 0 |
| | 20 | 373 | 6 | 18 | 72 | 277 | 0 | 0 |
| | 19 | 366 | 6 | 18 | 72 | 270 | 0 | 0 |
| 6 | 18 | 201 | 6 | 18 | 72 | 105 | 0 | 0 |
| | 17 | 192 | 6 | 18 | 72 | 96 | 0 | 0 |
| | 16 | 163 | 6 | 18 | 72 | 67 | 0 | 0 |
| 5 | 15 | 99 | 6 | 18 | 72 | 3 | 0 | 0 |
| | 14 | 50 | 6 | 18 | 26 | 0 | 0 | 0 |
| | 13 | 36 | 6 | 18 | 12 | 0 | 0 | 0 |
| 4 | 12 | 27 | 6 | 18 | 3 | 0 | 0 | 0 |
| | 11 | 12 | 6 | 6 | 0 | 0 | 0 | 0 |
| | 10 | 3 | 1 | 2 | 0 | 0 | 0 | 0 |
| 3 | 9 | 3 | 1 | 2 | 0 | 0 | 0 | 0 |
| | 8 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Image 00777, 250 Mbit/s

| Bit-plane | Code-Pass | Total | rlvl 0 | rlvl 1 | rlvl 2 | rlvl 3 | rlvl 4 | rlvl 5 |
|---|---|---|---|---|---|---|---|---|
| Σ | Σ | 48254 | 180 | 441 | 1525 | 5232 | 14648 | 26204 |
| 16 | *46* | *0* | *0* | *0* | *0* | *0* | *0* | *0* |
| 15 | *45* | *0* | *0* | *0* | *0* | *0* | *0* | *0* |
| | *44* | *0* | *0* | *0* | *0* | *0* | *0* | *0* |
| | *43* | *42* | *0* | *0* | *0* | *0* | *42* | *0* |
| 14 | *42* | *42* | *0* | *0* | *0* | *0* | *42* | *0* |
| | *41* | *42* | *0* | *0* | *0* | *0* | *42* | *0* |
| | *40* | *273* | *2* | *3* | *11* | *22* | *184* | *51* |
| 13 | *39* | *273* | *2* | *3* | *11* | *22* | *184* | *51* |
| | *38* | *273* | *2* | *3* | *11* | *22* | *184* | *51* |
| | *37* | *706* | *4* | *5* | *19* | *73* | *319* | *286* |
| 12 | *36* | *706* | *4* | *5* | *19* | *73* | *319* | *286* |
| | *35* | *706* | *4* | *5* | *19* | *73* | *319* | *286* |
| | *34* | *1601* | *6* | *7* | *25* | *99* | *598* | *866* |
| 11 | *33* | *1601* | *6* | *7* | *25* | *99* | *598* | *866* |
| | *32* | *1601* | *6* | *7* | *25* | *99* | *598* | *866* |
| | *31* | *3069* | *6* | *10* | *38* | *153* | *892* | *1970* |
| 10 | *30* | *3053* | *6* | *10* | *38* | *153* | *892* | *1954* |
| | *29* | *3052* | *6* | *10* | *38* | *153* | *892* | *1953* |
| | *28* | *4183* | *6* | *16* | *55* | *232* | *1003* | *2871* |
| 9 | *27* | *4130* | *6* | *16* | *55* | *232* | *1003* | *2818* |
| | *26* | *4053* | *6* | *16* | *55* | *232* | *1003* | *2741* |
| | *25* | *4004* | *6* | *16* | *69* | *280* | *1008* | *2625* |
| 8 | 24 | 3135 | 6 | 16 | 69 | 280 | 1008 | 1756 |
| | 23 | 2893 | 6 | 16 | 69 | 280 | 897 | 1625 |
| | 22 | 2661 | 6 | 18 | 72 | 288 | 897 | 1380 |
| 7 | 21 | 2176 | 6 | 18 | 72 | 288 | 896 | 896 |
| | 20 | 947 | 6 | 18 | 72 | 288 | 560 | 3 |
| | 19 | 655 | 6 | 18 | 72 | 288 | 268 | 3 |
| 6 | 18 | 384 | 6 | 18 | 72 | 288 | 0 | 0 |
| | 17 | 384 | 6 | 18 | 72 | 288 | 0 | 0 |
| | 16 | 384 | 6 | 18 | 72 | 288 | 0 | 0 |
| 5 | 15 | 384 | 6 | 18 | 72 | 288 | 0 | 0 |
| | 14 | 369 | 6 | 18 | 72 | 273 | 0 | 0 |
| | 13 | 174 | 6 | 18 | 72 | 78 | 0 | 0 |
| 4 | 12 | 96 | 6 | 18 | 72 | 0 | 0 | 0 |
| | 11 | 96 | 6 | 18 | 72 | 0 | 0 | 0 |
| | 10 | 34 | 6 | 18 | 10 | 0 | 0 | 0 |
| 3 | 9 | 24 | 6 | 18 | 0 | 0 | 0 | 0 |
| | 8 | 24 | 6 | 18 | 0 | 0 | 0 | 0 |
| | 7 | 24 | 6 | 18 | 0 | 0 | 0 | 0 |
| 2 | 6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

# Strategies

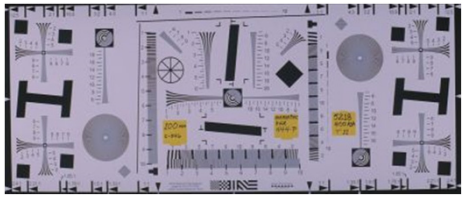
- kdu standard
  - Passes are skipped equally for all code-blocks
  - Some code-blocks are more important than others
- Block-Pass-Count-Relative
  - Relative to number of passes in the Code-Block
  - Blocks with many passes are truncated more heavily than blocks with few passes
- Subband-Relative
  - Relative to the bit-depth of Subband the Code-Block belongs to
  - Encoder’s PCRD-Opt is not overruled
- Image-Relative
  - Relative to the maximum bit-depth of all Subbands
  - low-frequency code-blocks are truncated more heavily
- Modified Strategies
  - prevent skipping code-passes for certain subbands

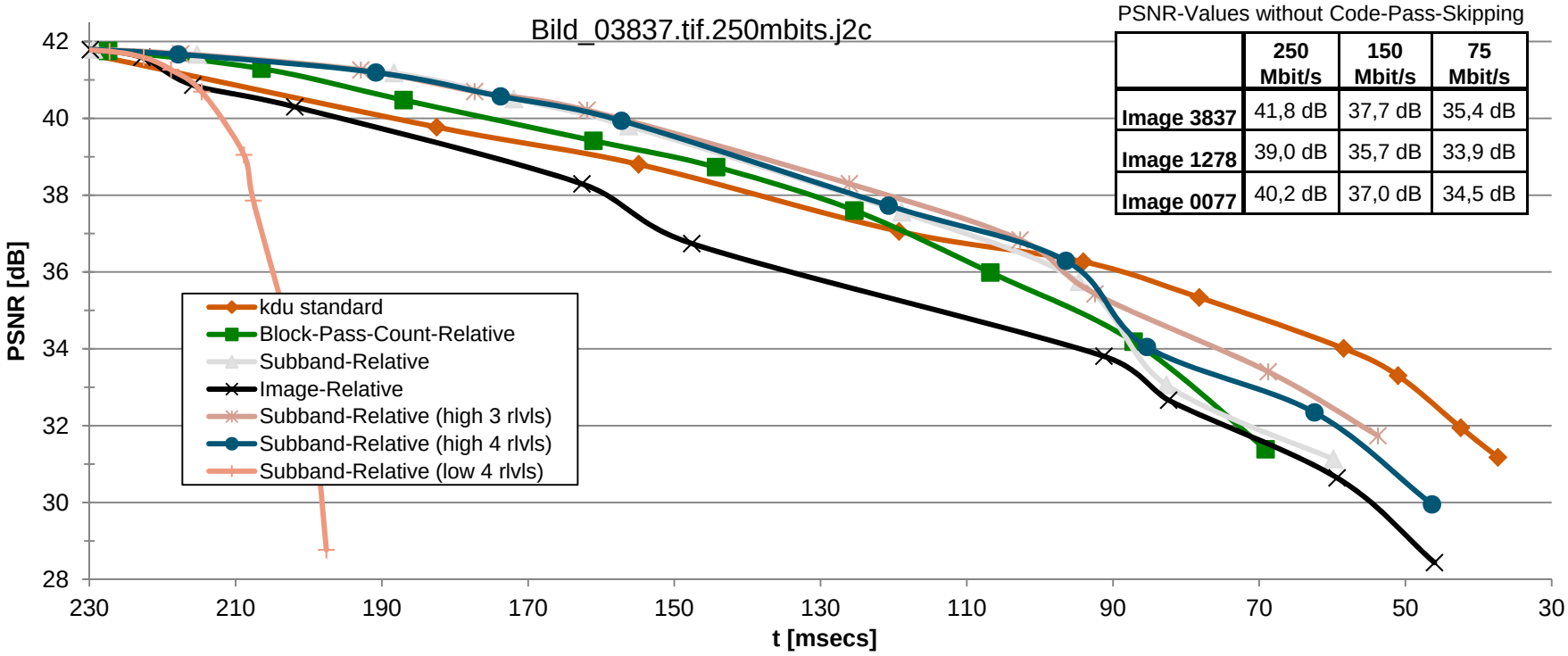
# Results

Image 3837

Image 1278


Image 77





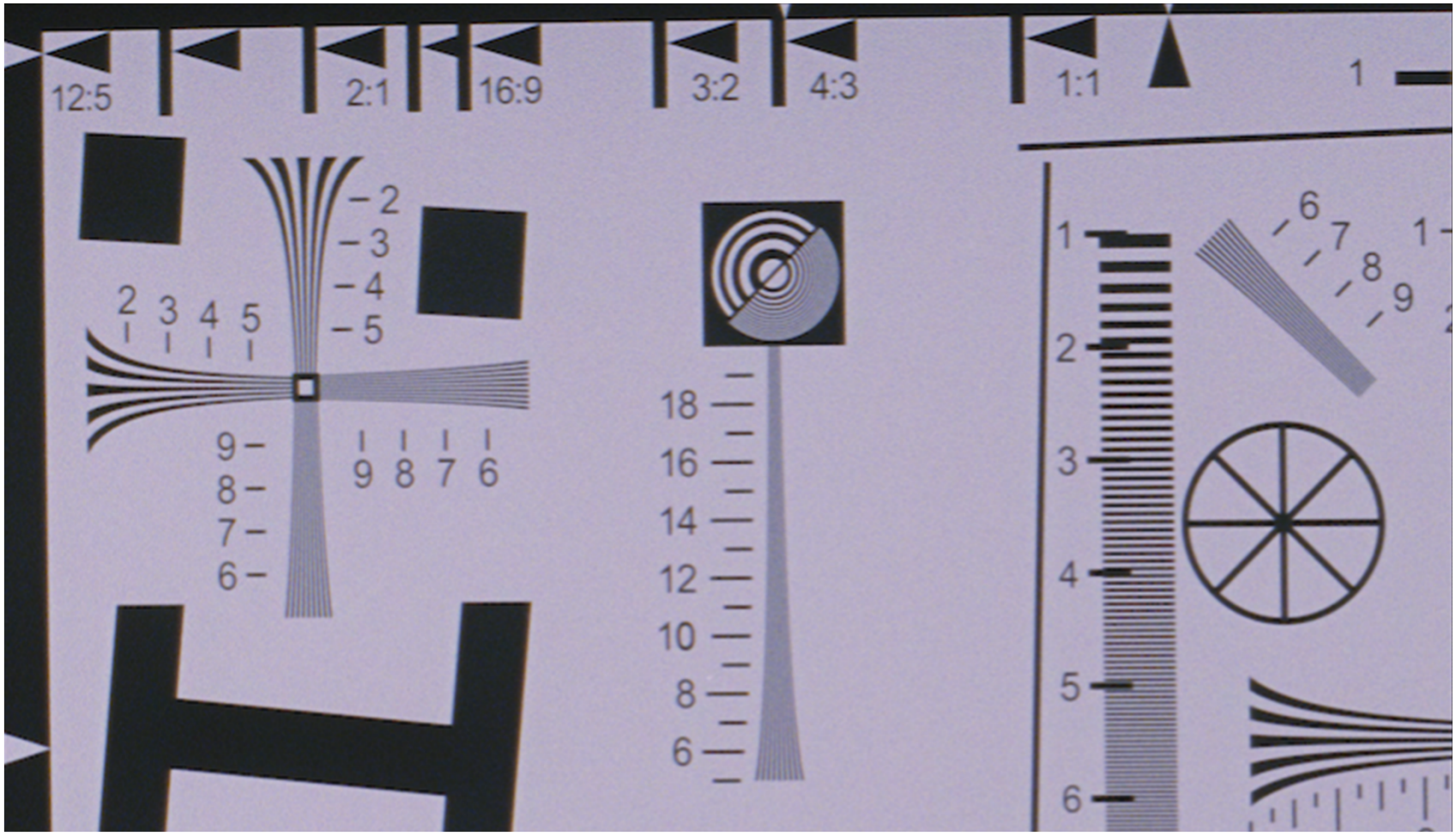

PSNR-Values without Code-Pass-Skipping

| | 250 Mbit/s | 150 Mbit/s | 75 Mbit/s |
|---|---|---|---|
| **Image 3837** | 41,8 dB | 37,7 dB | 35,4 dB |
| **Image 1278** | 39,0 dB | 35,7 dB | 33,9 dB |
| **Image 0077** | 40,2 dB | 37,0 dB | 34,5 dB |



Fraunhofer IIS

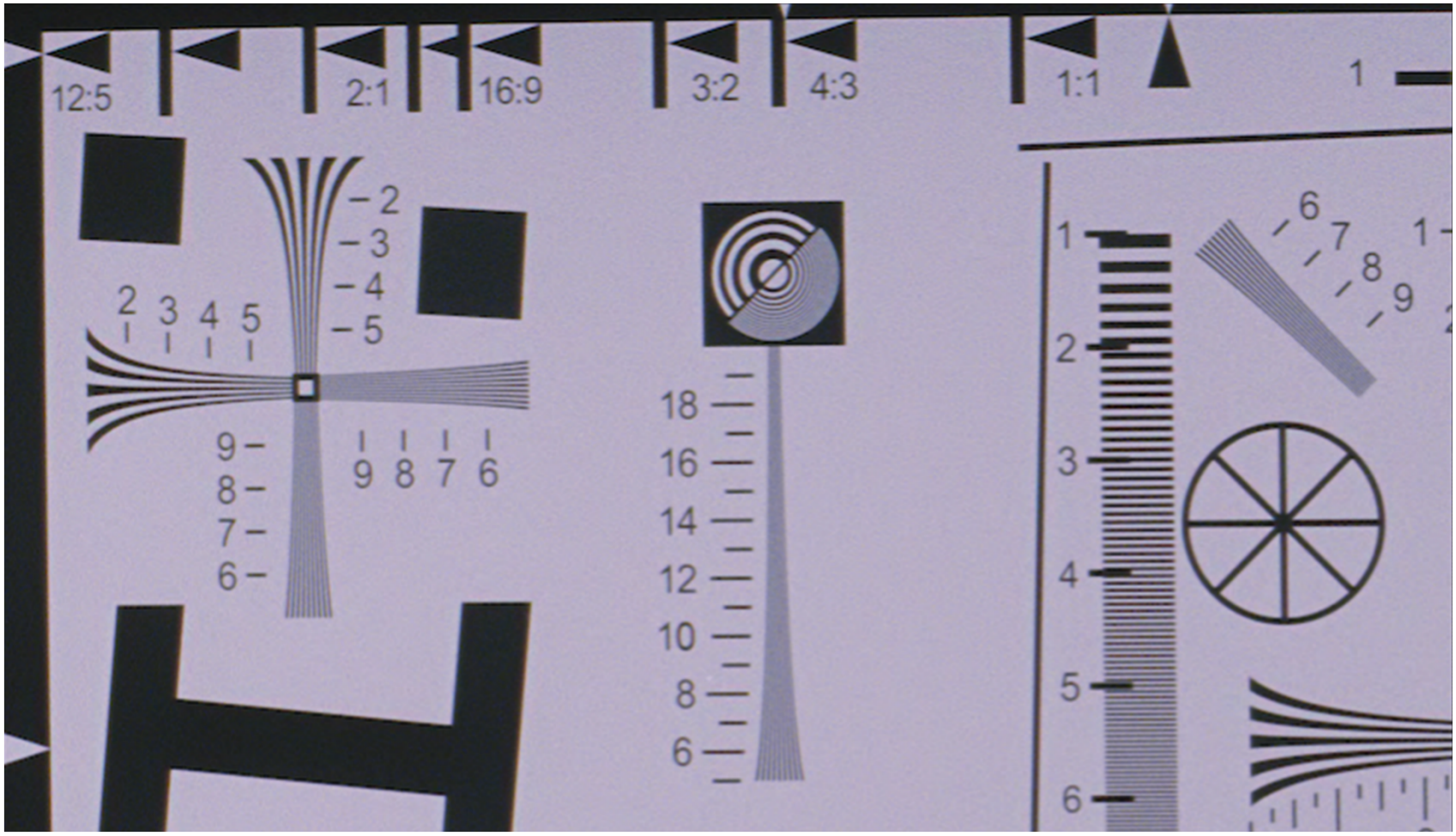
# Quality Index 100






Fraunhofer IIS

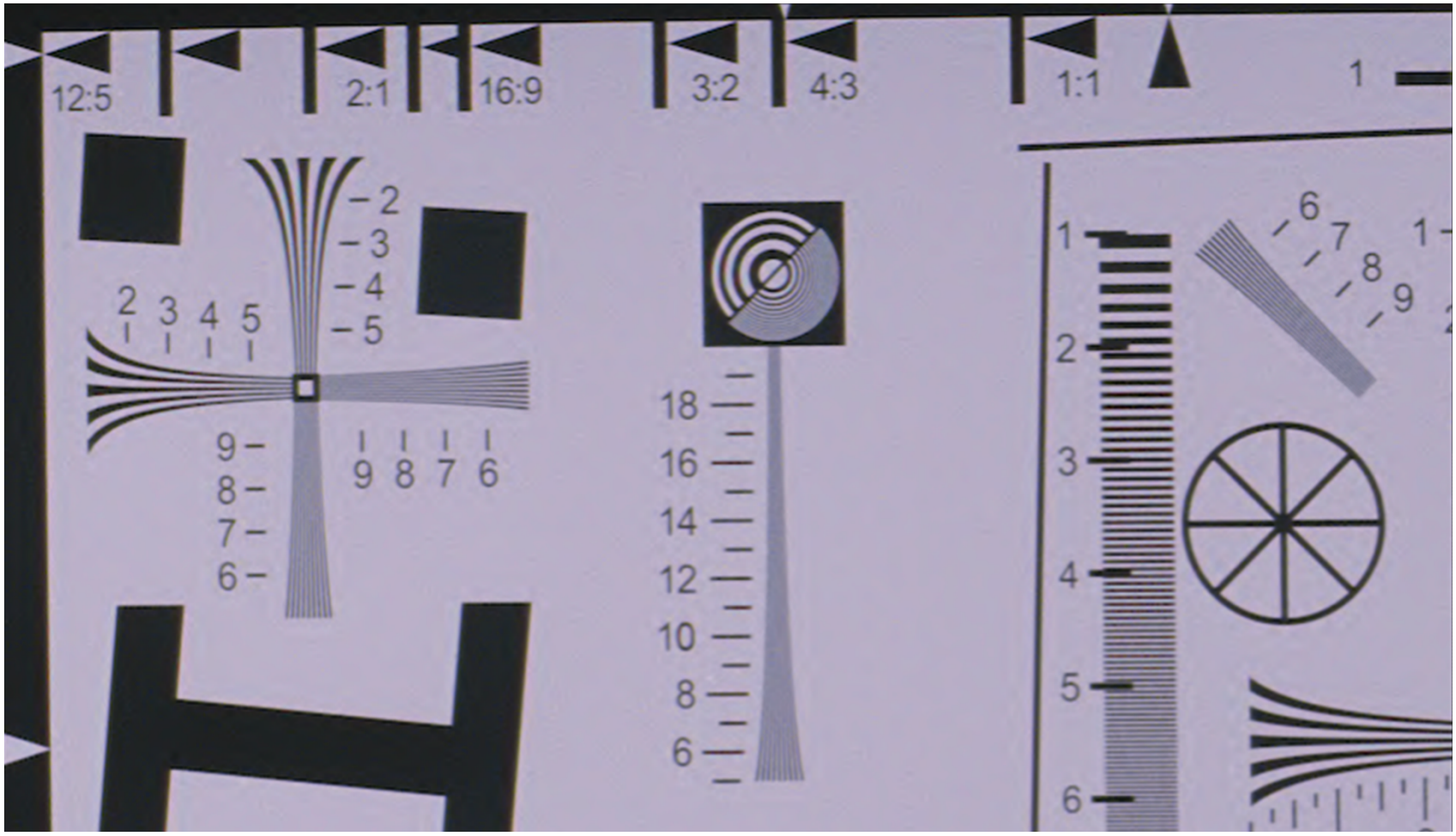
# Quality Index 50






Fraunhofer IIS

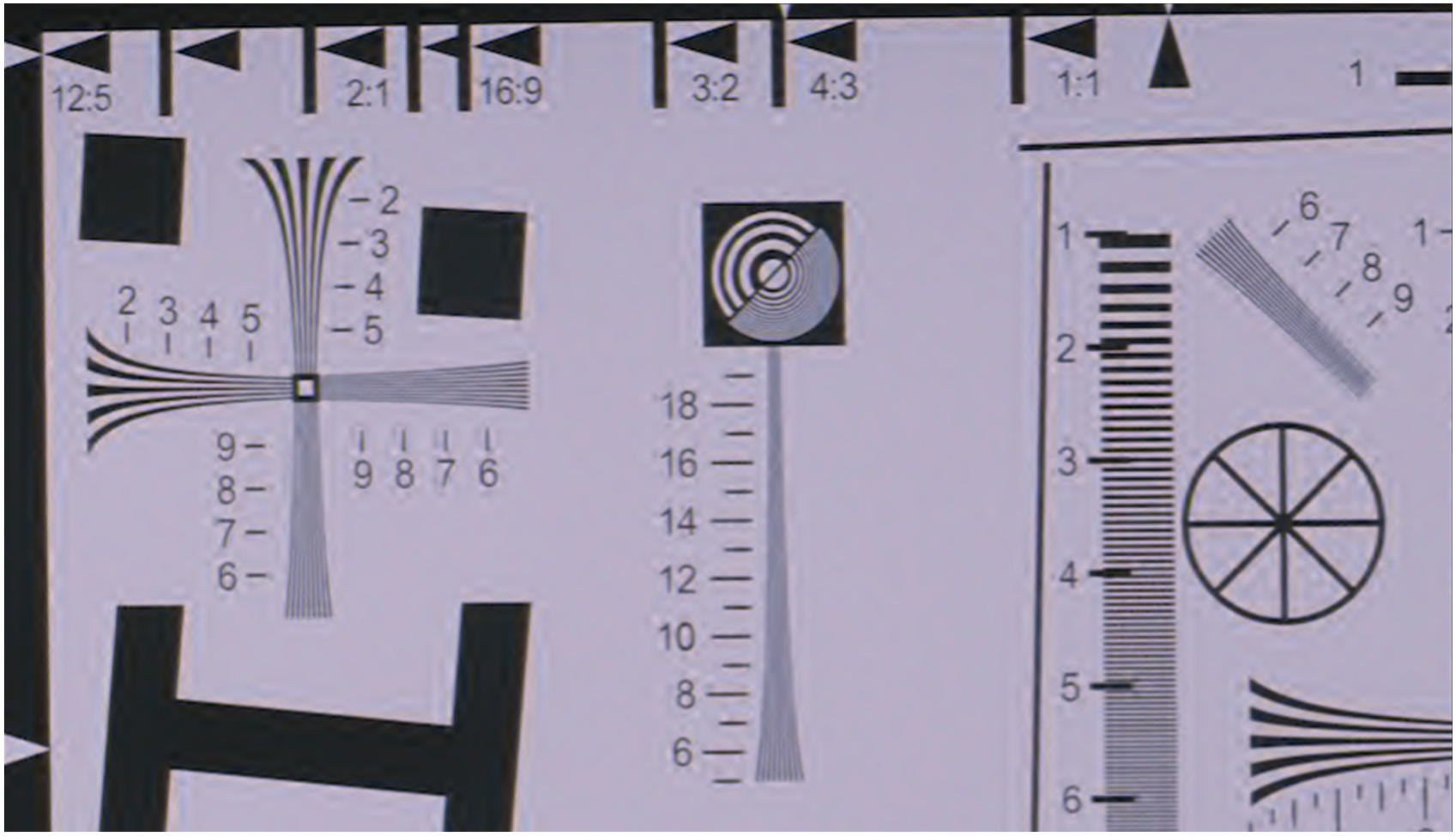
# Quality Index 25






Fraunhofer IIS

# Quality Index 0

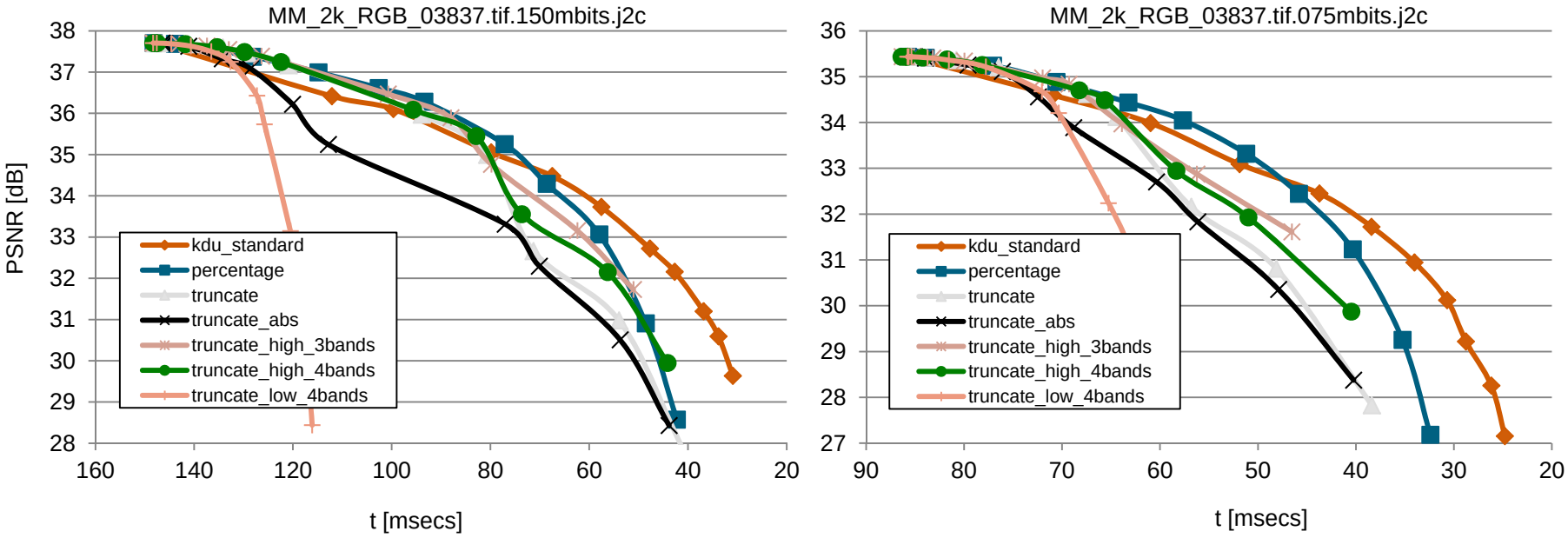





Fraunhofer IIS

# Results (cont.)

- Similar Results for 150 Mbit/s and 75 Mbit/s @ 24 fps






Fraunhofer IIS

# Conclusion

- Significant speed-up at quality losses that are tolerable in many applications
- "Subband-Relative“-strategy performs best
  - same speed-up, lower quality-loss
  - compared to reference from *Kakadu Software v6*
- Decision on what quality loss is still tolerable can be forwarded to the user
  - Quality-vs.-Performance Slider
  - Doubling the speed costs 3-4 dB
- Curve initially stays almost level
  - significant speed-up at almost no cost when reconstructing only a 24-bit preview from a 36-bit source (e.g. DCP)

# Thank you for your Attention


- volker.bruns@iis.fraunhofer.de