AVIF

Mini-Project Presentation

Pierre Dupuy

15 December 2025

EPFL

Table of Contents

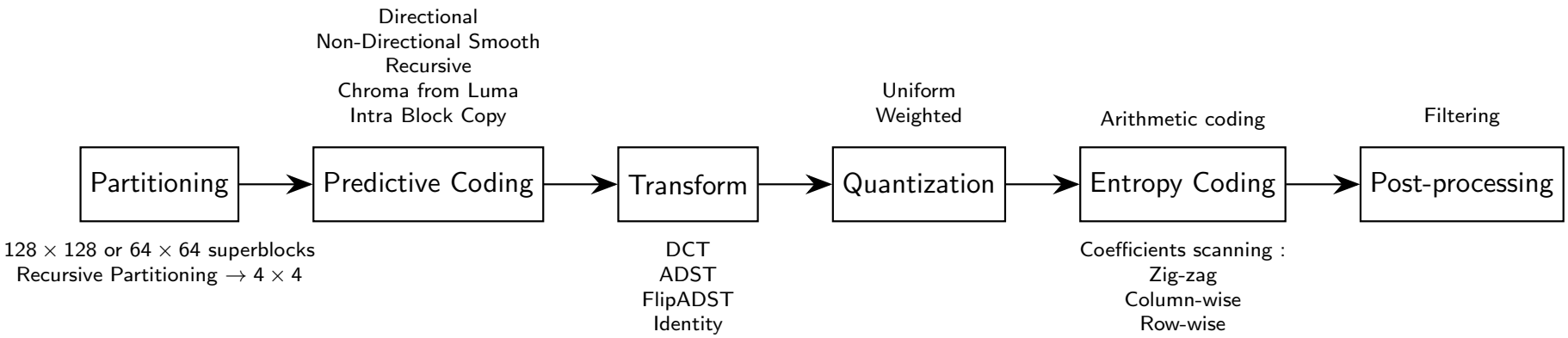
EPFL

Background

- 2019
- Alliance for Open Media AOM (Amazon, Apple, Cisco, Google, Microsoft, Mozilla, Netflix,...)
- Side effect of AV1
- HEIF container using AV1 instead of H.265 (HEIC)
- AVIF and WebP are "cousins"
  - VP8 $\rightarrow$ WebP
  - VP9 $\rightarrow$ AV1 $\rightarrow$ AVIF

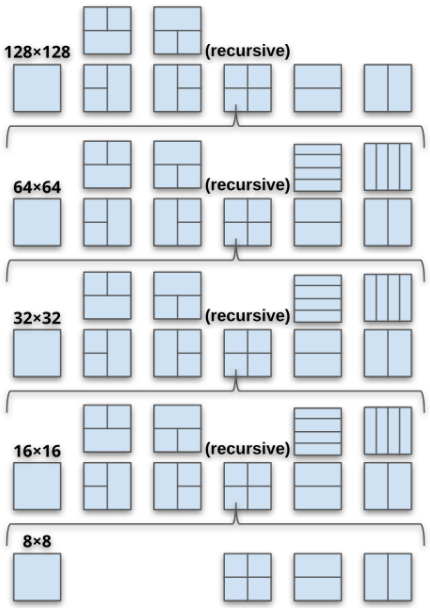
EPFL

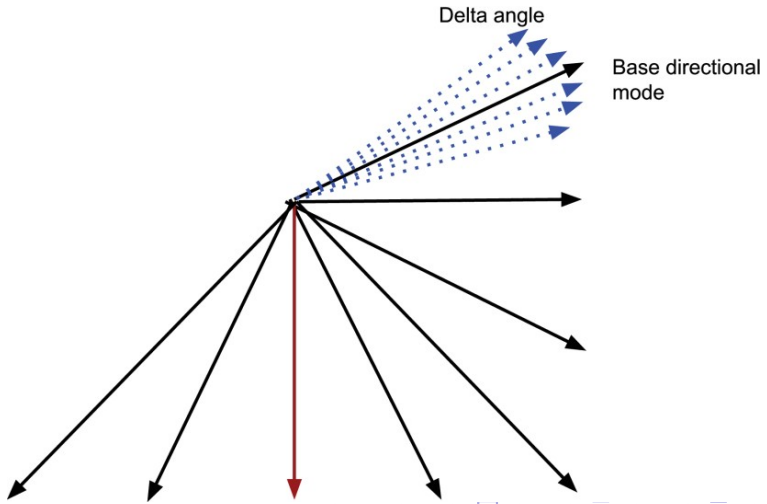
Block Diagram



Highlights (1/2)

- Partitioning
- Prediction
  - 56 directional modes
  - 6 non-directional modes
    - DC, Smooth, Smooth H, Smooth V, Paeth, Palette
  - 5 recursive filtering modes
    - DC, Vertical, Horizontal, 157°, Paeth
  - 1 chroma-only mode (Chroma from Luma)
  - Intra block copy





Highlights (2/2)

- Transforms
  - 16 types (DCT, ADST, FlipADST, Identity)×(DCT, ADST, FlipADST, Identity)
  - Size up to 64×64
  - Rectangular transforms
  - Recursive quad-tree transform size splitting
- Filtering
  - Deblocking Filter
  - Constrained Directional Enhancement Filter (Ringing artifacts)
  - Loop Restoration filter
    - Wiener filter
    - Self-Guided Filter
- YUV 4:0:0 (monochrome), 4:2:0, 4:2:2, **4:4:4**, RGB
- 8, 10 and 12-bit depth color (HDR)
- (Lossy) α channel (transparency)
- Optional downscaling/upscaling ("Superresolution")

EPFL

libavif

- AOM library
- C implementation
- `.\avifenc -q Q -s S -y Y input.[jpg|png|y4m] output.avif`
- variation in -q to change the BPP and fine tuning with -y (420, 422 or 444)
- -s put at 4

EPFL

Table of Contents

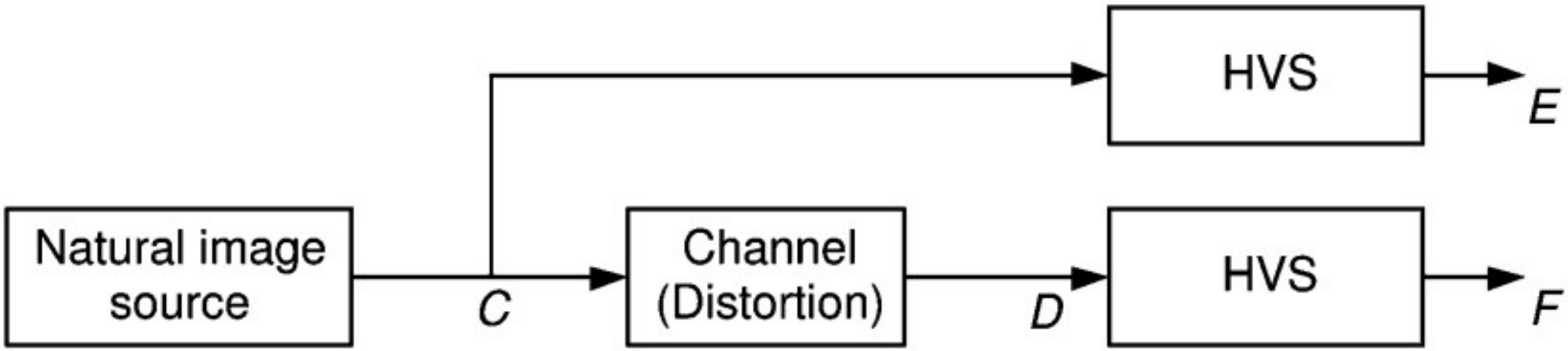
EPFL

VIF : Visual Information Fidelity (1/4)

- Laboratory of Image and Video Engineering (LIVE), University of Texas, 2006
- Full reference
- *Objective* QA that is consistent with *Subjective* human evaluation
- Based on a HVS and distortion channel model
- Natural images in the wavelet domain using Gaussian Scale Mixtures (GSMs)
- PIQ Library implementation (Python)

EPFL

VIF : Visual Information Fidelity (2/4)



$\mathcal{C} = \mathcal{S} \cdot \mathcal{U} = \{S_i \cdot \vec{U}_i : i \in I\}$ I set of spacial indices for the RF[1], $\mathcal{S}$ RF of positive scalars, $\mathcal{U}$ Gaussian vector RF (mean zero and covariance $\boldsymbol{C_U}$). $\vec{U}_i$ independent of $\vec{U}_j$, $\forall i \neq j$

$\mathcal{D} = \mathcal{GC} + \mathcal{V} = \{g_i \vec{C}_i + \vec{V}_i : i \in I\}$ $\mathcal{G}$ deterministic scalar gain field, $\mathcal{V}$ AGW[2] RF with variance $\boldsymbol{C_V} = \sigma_v^2 \boldsymbol{I}$

$\mathcal{E} = \mathcal{C} + \mathcal{N}$ (reference image) $\vec{N}_i$, $\vec{N}'_i$ zero-mean uncorrelated multivariate Gaussian

$\mathcal{F} = \mathcal{D} + \mathcal{N}'$ (test image) $\mathcal{N}, \mathcal{N}'$ independent of $\mathcal{U}, \mathcal{S}, \mathcal{V}$. Covariance $\boldsymbol{C_N} = \boldsymbol{C_{N'}} = \sigma_n^2 \boldsymbol{I}$

[1] RF = Random Field

[2] Additive zero-mean Gaussian Noise

VIF : Visual Information Fidelity (3/4)

$\overrightarrow{C}^N = (\overrightarrow{C_1}, \overrightarrow{C_2}, ..., \overrightarrow{C_N})$ denote N elements from $\mathcal{C}$.

S^N, $\overrightarrow{D}^N$, $\overrightarrow{E}^N$, $\overrightarrow{F^N}$ correspondingly defined

Mutual information $I(\overrightarrow{C}^N; \overrightarrow{E}^N | S^N = s^N)$ information extracted from HVS output by the brain when a test image viewed for a realization of S^N

We assume $\mathcal{G}$, σ_v^2, σ_n^2 known, $\boldsymbol{C}_U = \boldsymbol{Q}\boldsymbol{\Lambda}\boldsymbol{Q}^T$, $\boldsymbol{Q}$ orthonormal, $\boldsymbol{\Lambda}$ diagonal with eigenvalues λ_k

EPFL

VIF : Visual Information Fidelity (4/4)

$$I(\overrightarrow{C}^N; \overrightarrow{E}^N | S^N = s^N) = \frac{1}{2} \sum_{i=1}^{N} \sum_{k=1}^{M} \log_2(1 + \frac{s_i^2 \lambda_k}{\sigma_n^2})$$

$$I(\overrightarrow{C}^N; \overrightarrow{F}^N | S^N = s^N) = \frac{1}{2} \sum_{i=1}^{N} \sum_{k=1}^{M} \log_2(1 + \frac{g_i^2 s_i^2 \lambda_k}{\sigma_v^2 + \sigma_n^2})$$

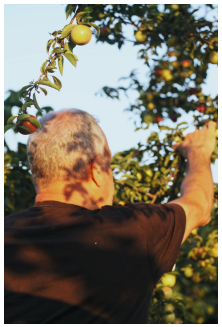
$$VIF = \frac{\sum_{j \in \text{subband}} I(\overrightarrow{C}^{N,j}; \overrightarrow{F}^{N,j} | s^{N,j})}{\sum_{j \in \text{subband}} I(\overrightarrow{C}^{N,j}; \overrightarrow{E}^{N,j} | s^{N,j})}$$

EPFL

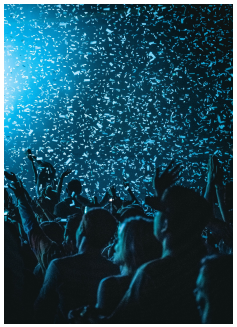
Dataset

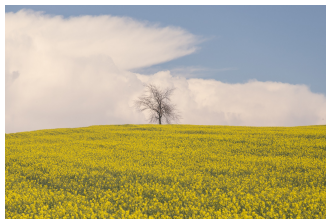




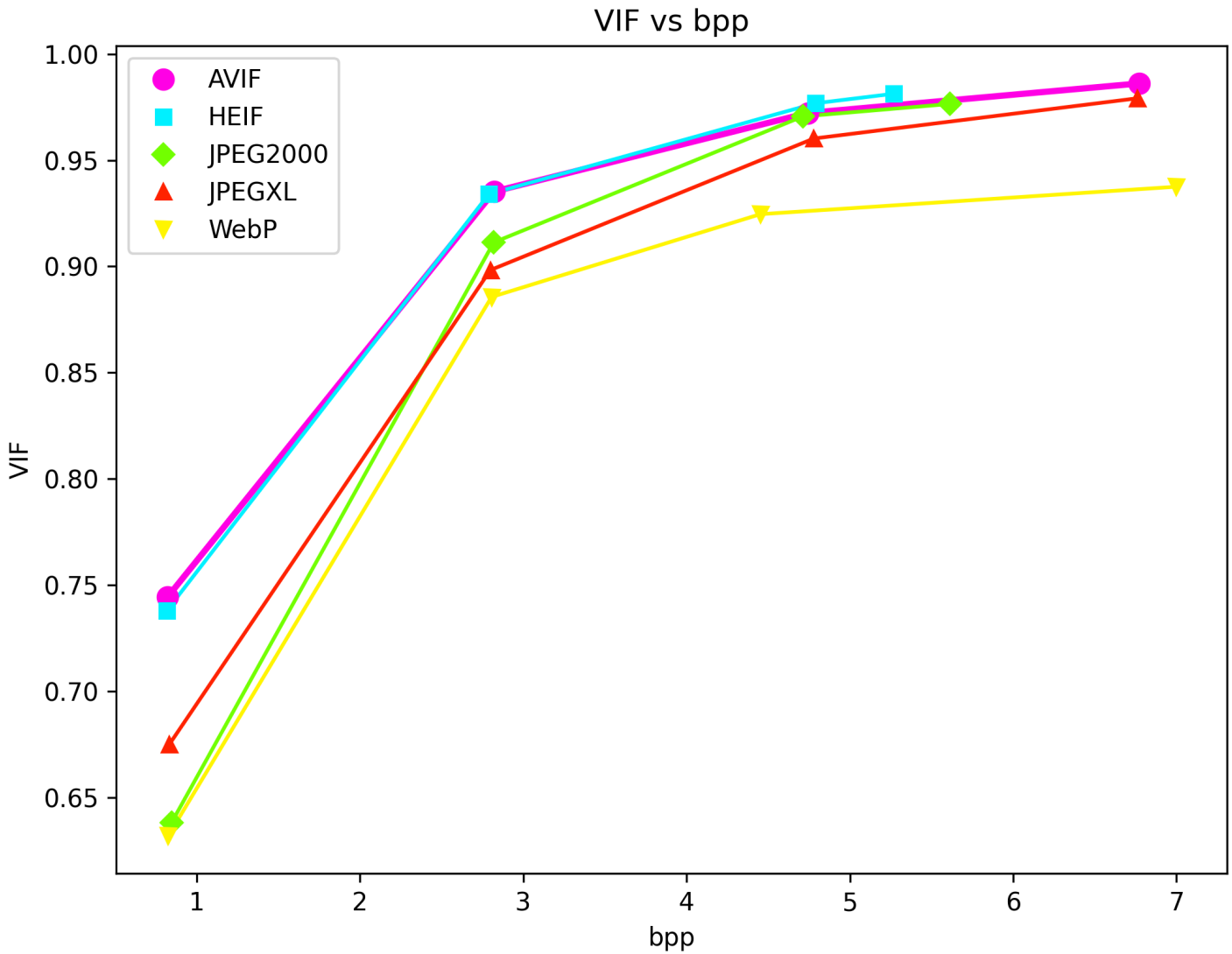


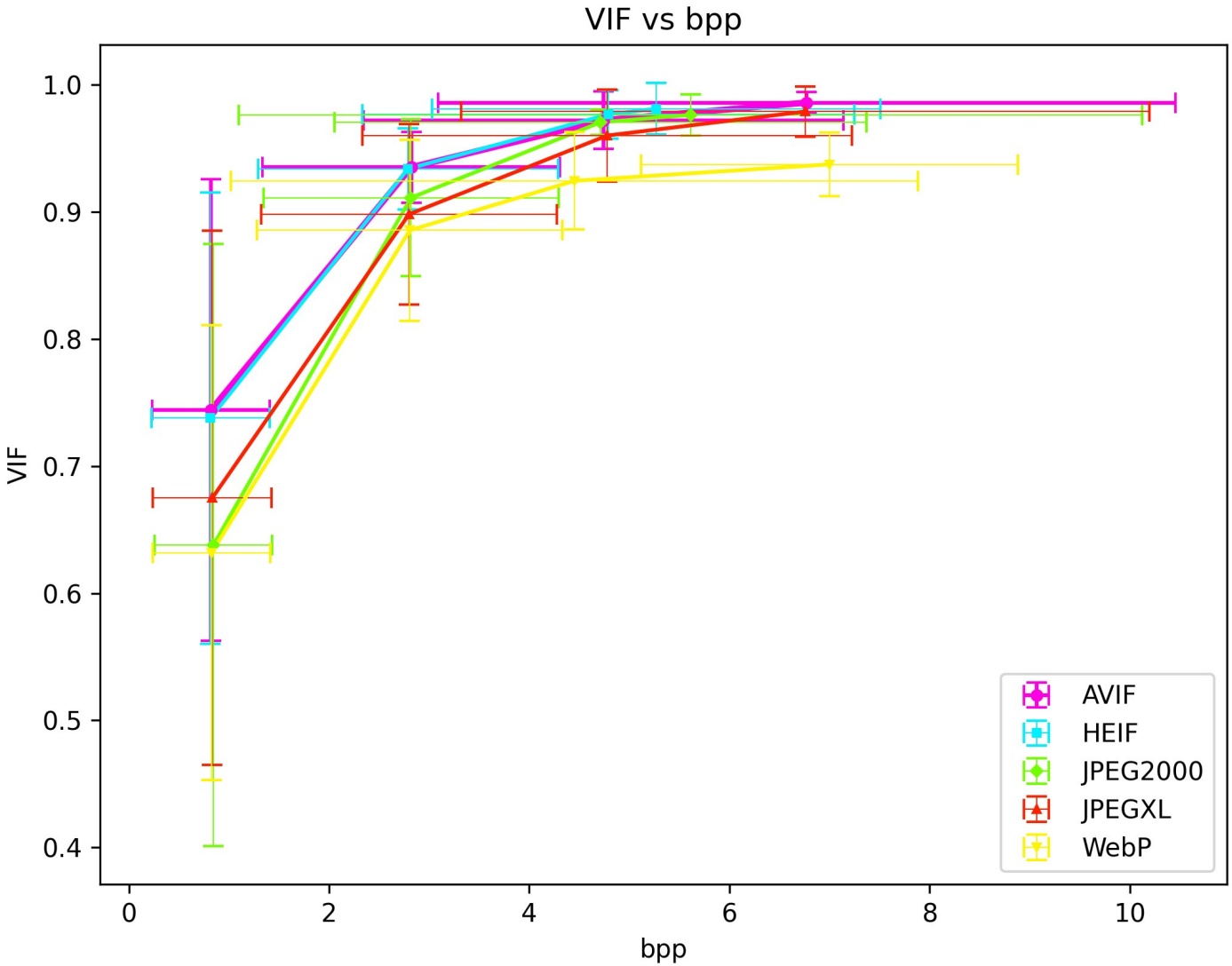


EPFL

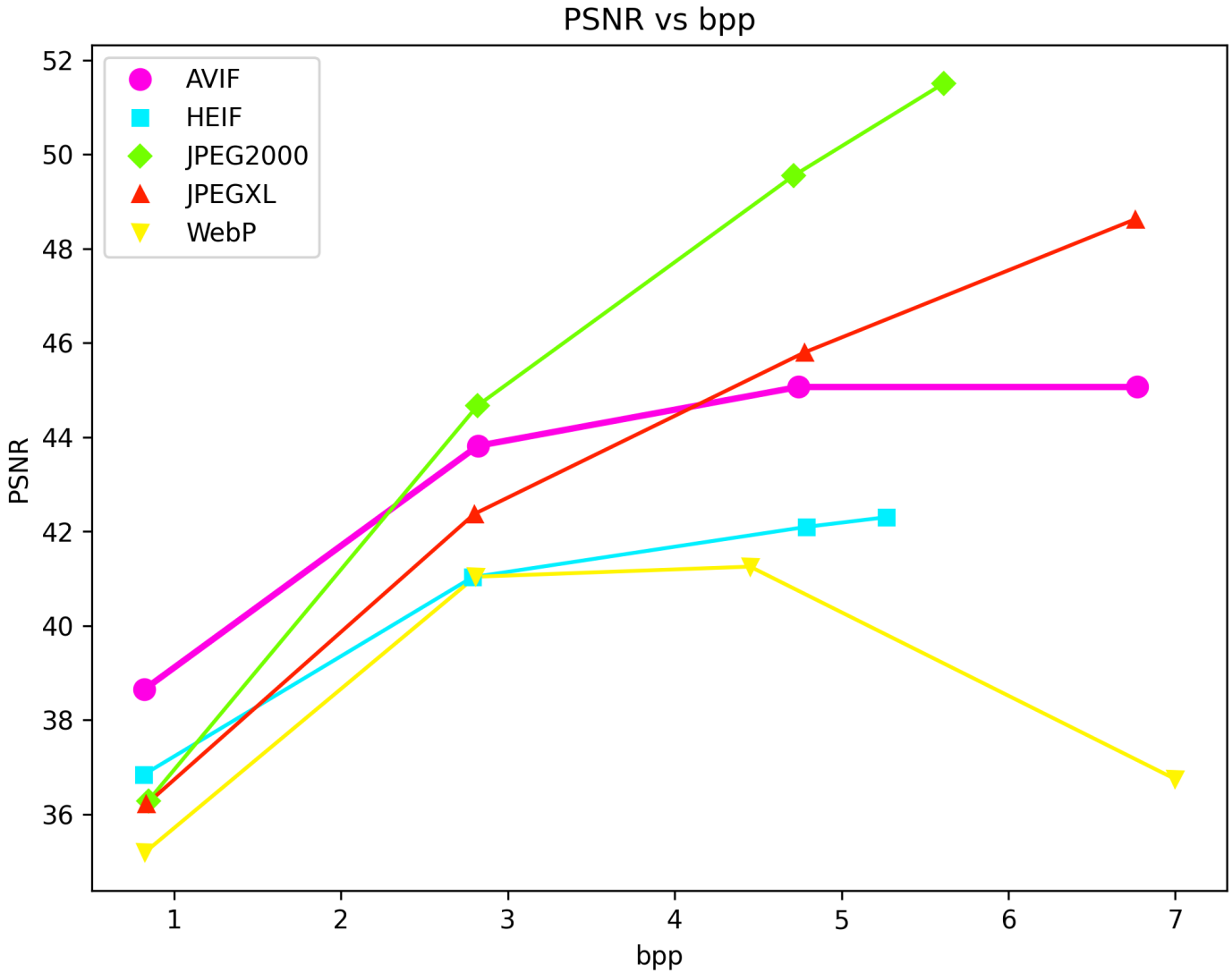
Measurements results : VIF



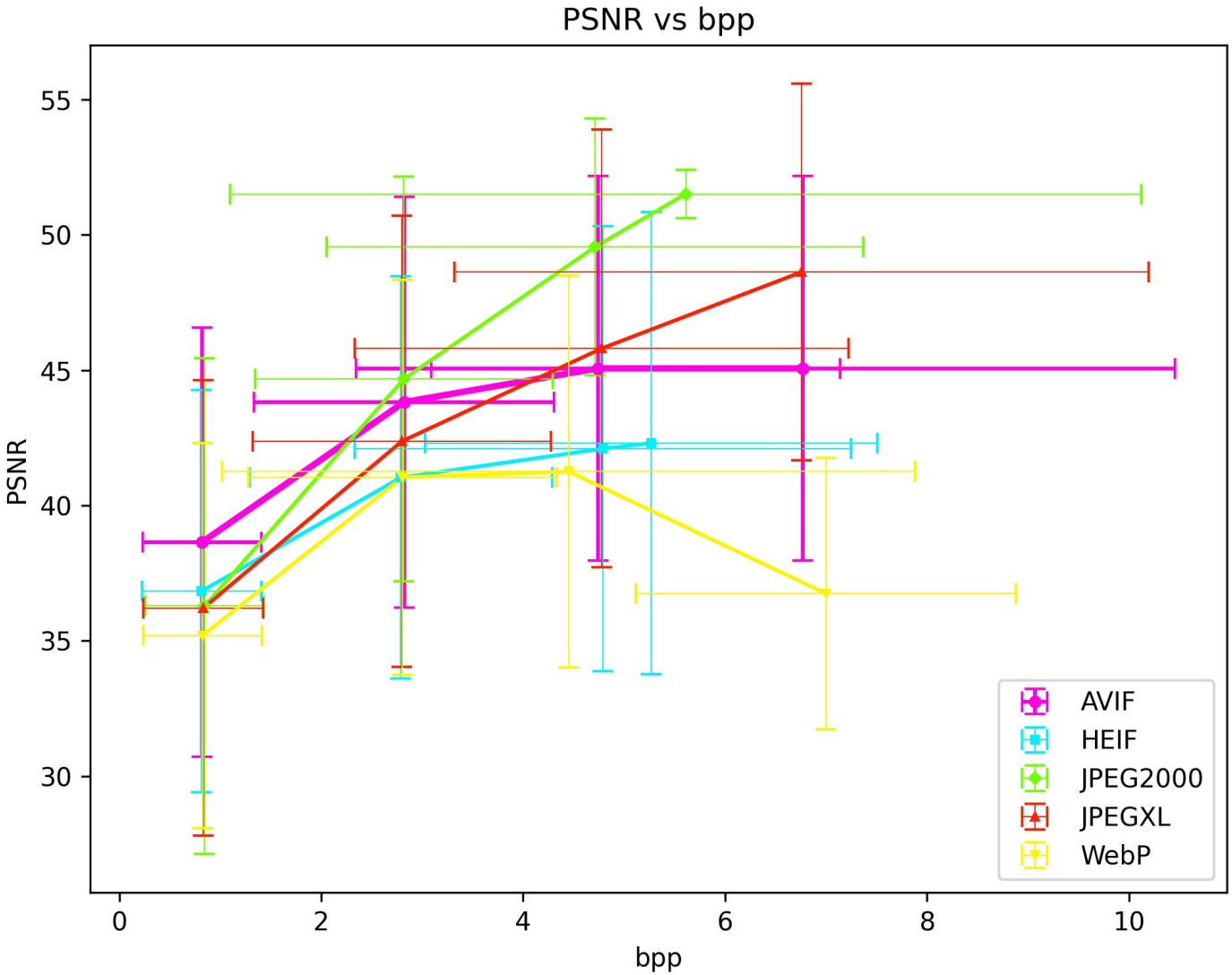
Measurement results : VIF (with error bars)



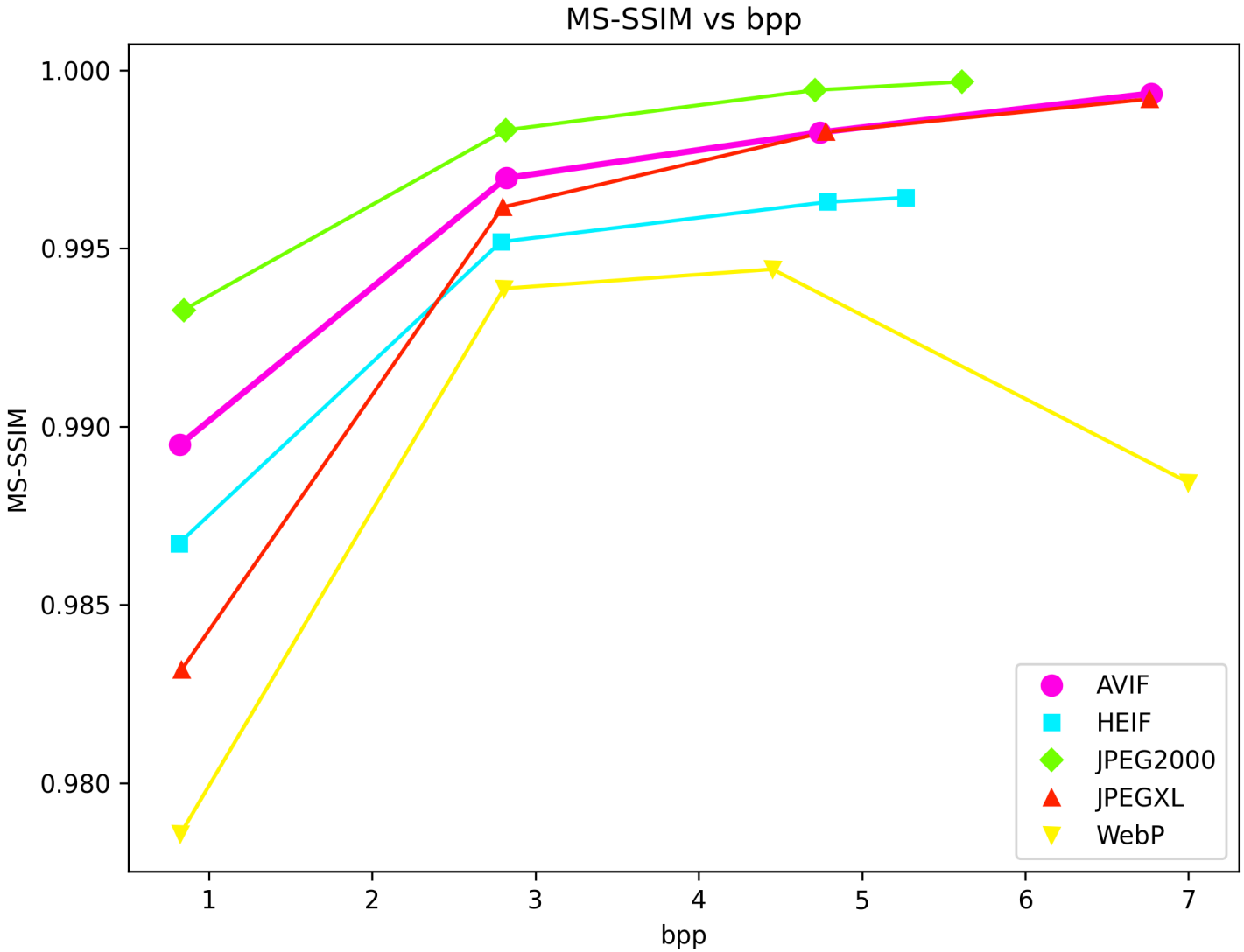
Measurements results : PSNR



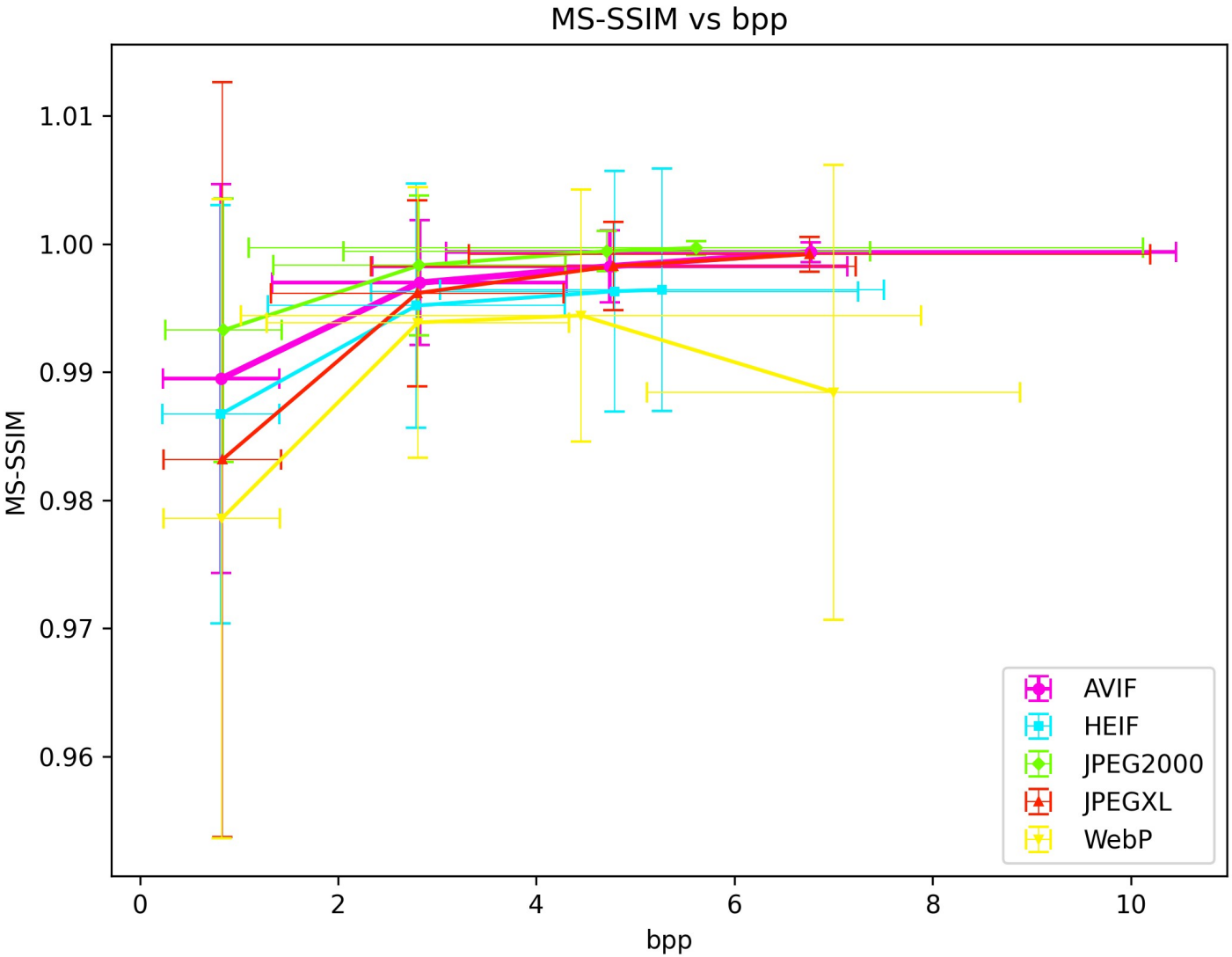
Measurements results : PSNR (with error bars)



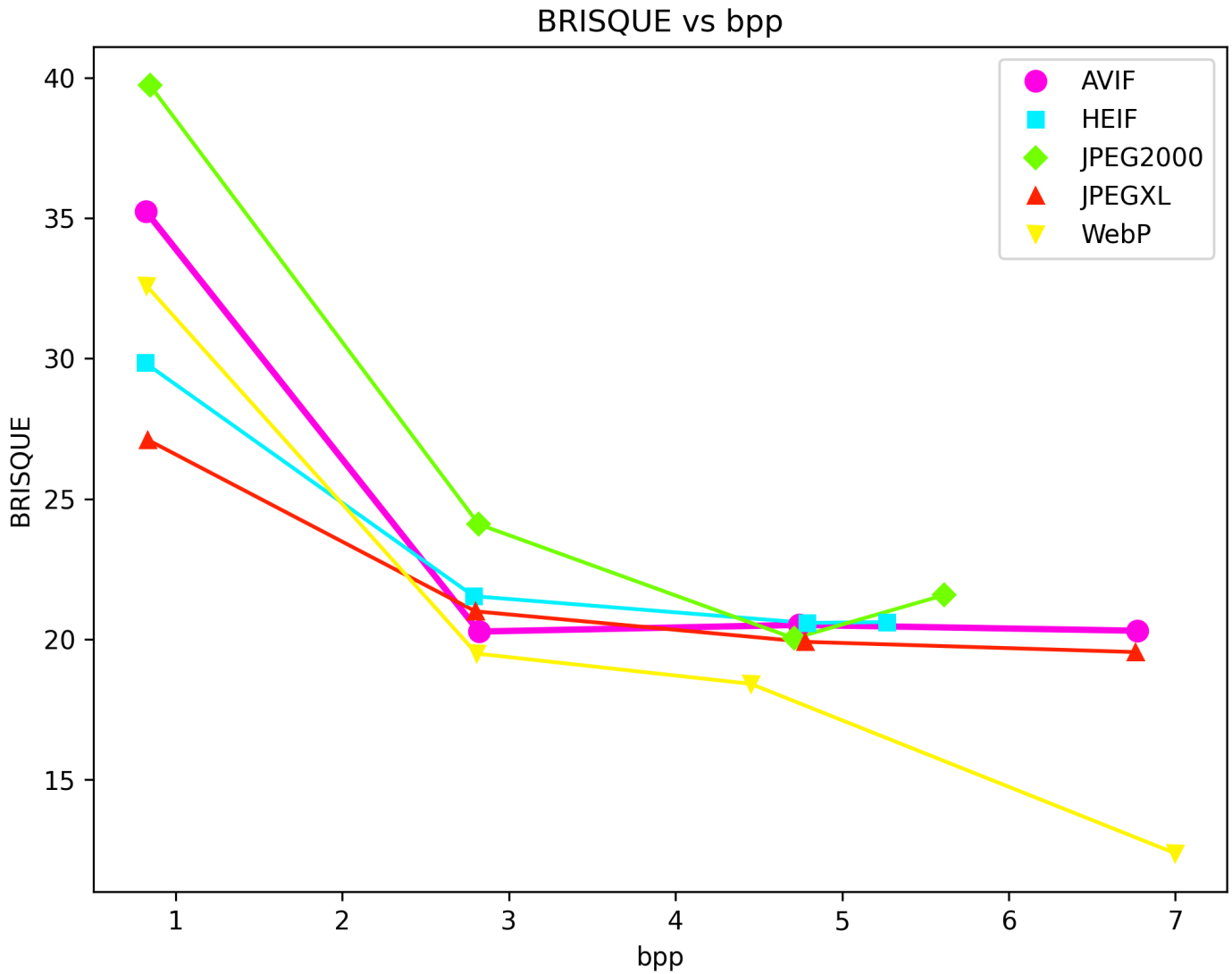
Measurements results : MS-SSIM



Measurements results : MS-SSIM (with error bars)



Measurements results : BRISQUE



Measurements results : BRISQUE (with error bars)

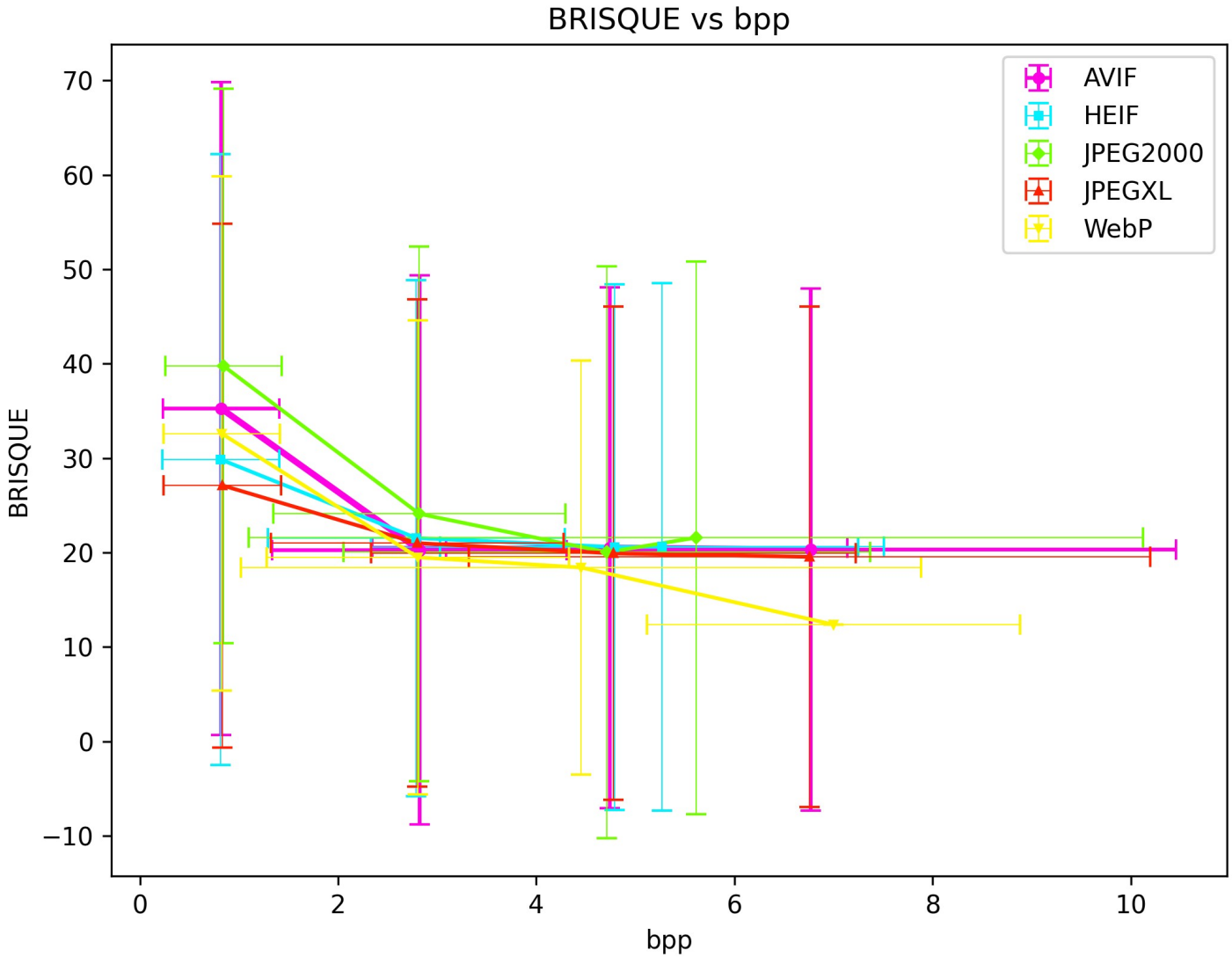


Table of Contents

EPFL

Context

- Large range of devices used for streaming
- The browsing of the catalog is made through the thumbnails
- User Interface very rich in image content $\rightarrow$ need of good compression
- Image assets are a mix of photograph, text, drawings, etc...
- Hard edges, chroma differences, etc...

EPFL

Study

- CLIC Dataset
  - 303 images
  - 2048x1320 resolution
- JPEG with standard quantization matrix as reference

`jpeg-mse`	JPEG with a flat quantization matrix that minimizes MSE
`jpeg-ms-ssim`	JPEG with a quantization matrix tuned to maximize MS-SSIM
`jpeg-im`	JPEG with a quantization matrix recommended by ImageMagick
`jpeg-hvs-psnr`	JPEG with a quantization matrix tuned to maximize HVS-PSNR
`webp`	Webp codec
`kakadu-mse`	Implementation of JPEG 2000 by Kakadu; a setting that minimizes MSE
`kakadu-visual`	Implementation of JPEG 2000 by Kakadu; a setting that applies weighted quantization to maximize visual quality
`openjpeg`	Implementation of JPEG 2000 in libopenjpeg
`hevc`	HEVC intra-frame coding implemented in HM reference software
`avif-mse`	AVIF implemented in libaom; a setting that minimizes MSE
`avif-ssim`	AVIF implemented in libaom; a setting that maximizes SSIM

EPFL

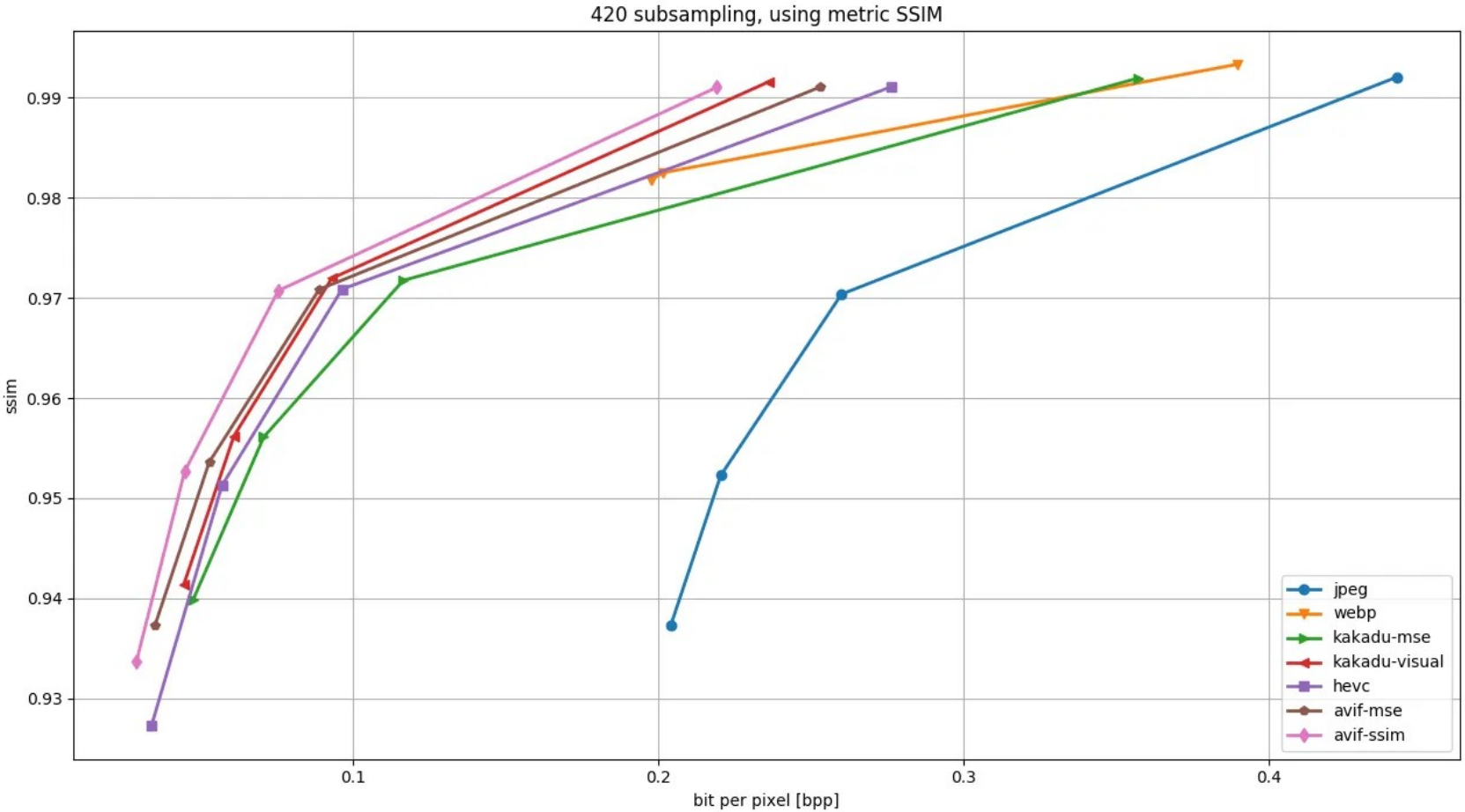
Results (1/3)

420 subsampling	Mean BD rate with respective metric [percentage]				
	SSIM	MS_SSIM	VIF	PSNR_Y	PSNR_AVG
jpeg-mse	37.82	27.90	0.70	-6.10	-6.15
jpeg-ms-ssim	-4.91	2.33	3.73	-0.21	-3.08
jpeg-im	-3.47	1.06	2.54	0.79	-1.84
jpeg-hvs-psnr	-13.65	-3.44	4.04	4.08	1.66
webp	-19.71	-18.00	-28.06	-36.93	-35.96
kakadu-mse	-27.22	-31.65	-39.91	-48.00	-46.32
kakadu-visual	-46.89	-48.16	-31.91	-32.51	-32.12
openjpeg	-6.35	-22.64	-34.31	-40.65	-38.14
hevc	-38.35	-40.63	-52.77	-59.32	-57.92
avif-mse	-43.28	-45.21	-56.70	-62.76	-62.46
avif-ssim	-47.91	-48.99	-56.61	-61.56	-61.16

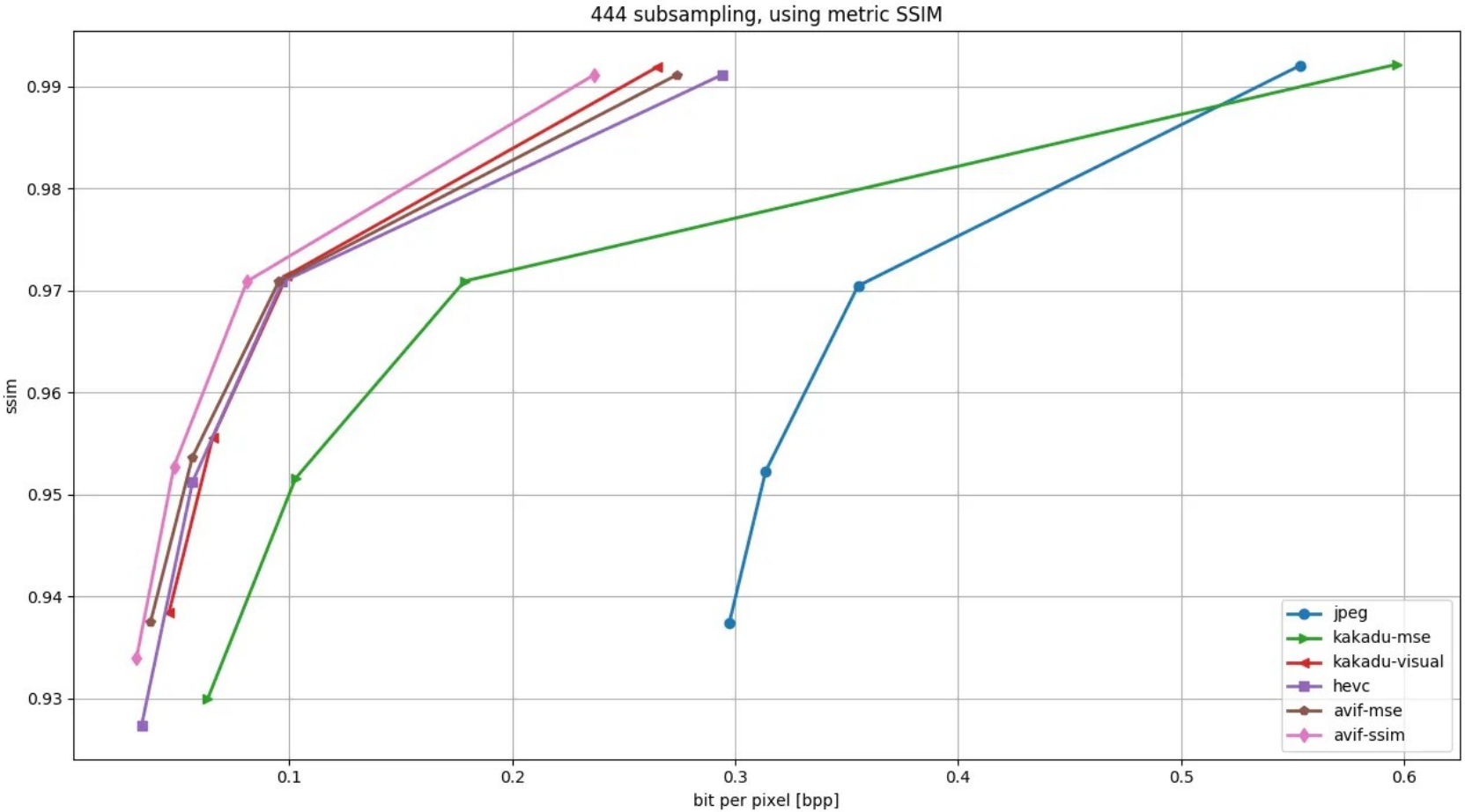
444 subsampling	Mean BD rate with respective metric [percentage]				
	SSIM	MS_SSIM	VIF	PSNR_Y	PSNR_AVG
jpeg-mse	34.01	24.79	0.25	-5.92	-4.60
jpeg-ms-ssim	3.18	10.32	12.77	8.69	-0.77
jpeg-im	1.04	5.62	8.36	6.65	-1.18
jpeg-hvs-psnr	-8.68	1.19	9.26	9.28	1.31
kakadu-mse	-5.05	-9.61	-17.91	-29.31	-27.45
kakadu-visual	-54.73	-55.69	-40.98	-41.51	-42.66
openjpeg	14.58	-4.63	-16.45	-23.95	-17.19
hevc	-48.21	-50.03	-59.31	-64.93	-66.00
avif-mse	-51.26	-52.67	-61.83	-67.16	-68.98
avif-ssim	-55.10	-55.91	-61.83	-66.20	-67.96

EPFL

Results (2/3)


420 subsampling, using metric SSIM
ssim
0.99
0.98
0.97
0.96
0.95
0.94
0.93
0.1
0.2
0.3
0.4
bit per pixel [bpp]
jpeg
webp
kakadu-mse
kakadu-visual
hevc
avif-mse
avif-ssim

Results (3/3)


444 subsampling, using metric SSIM
ssim
0.99
0.98
0.97
0.96
0.95
0.94
0.93
0.1
0.2
0.3
0.4
0.5
0.6
bit per pixel [bpp]
jpeg
kakadu-mse
kakadu-visual
hevc
avif-mse
avif-ssim

Bibliography I

[1] *Alliance for Open Media*. en. Page Version ID: 1315967256. Oct. 2025. URL: https://en.wikipedia.org/w/index.php?title=Alliance_for_Open_Media&oldid=1315967256 (visited on 12/14/2025).

[2] linux.conf.au. *The AV1 Video Codec*. Jan. 2019. URL: https://www.youtube.com/watch?v=qubPzBcYCTw (visited on 12/12/2025).

[3] *AV1*. en. Page Version ID: 1325989913. Dec. 2025. URL: https://en.wikipedia.org/w/index.php?title=AV1&oldid=1325989913 (visited on 12/12/2025).

[4] *AVIF*. en. Page Version ID: 1325485271. Dec. 2025. URL: https://en.wikipedia.org/w/index.php?title=AVIF&oldid=1325485271 (visited on 12/12/2025).

EPFL

Bibliography II

[5] Netflix Technology Blog. *AVIF for Next-Generation Image Coding*. en. Feb. 2020. URL: https://netflixtechblog.com/avif-for-next-generation-image-coding-b1d75675fe4 (visited on 12/11/2025).

[6] *AOMediaCodec/libavif*. original-date: 2019-06-10T17:54:32Z. Dec. 2025. URL: https://github.com/AOMediaCodec/libavif (visited on 12/11/2025).

[7] Henri Helvetica. *Avif For Next Generation Image Coding by Aditya Mavlankar [IMAGE READY]*. Nov. 2020. URL: https://www.youtube.com/watch?v=5RX6IgIF8bw (visited on 12/11/2025).

EPFL

Bibliography III

[8] Henri Helvetica. *The AVIF Image Format by Kornel Lesiński [IMAGE READY]*. Nov. 2020. URL: https://www.youtube.com/watch?v=VHm5Ql33JYw (visited on 12/11/2025).

[9] Jingning Han et al. "A Technical Overview of AV1". In: *Proceedings of the IEEE* 109.9 (Sept. 2021), pp. 1435–1462. ISSN: 1558-2256. DOI: 10.1109/JPROC.2021.3058584. URL: https://ieeexplore.ieee.org/document/9363937/figures (visited on 12/11/2025).

[10] H.R. Sheikh and A.C. Bovik. "Image information and visual quality". In: *IEEE Transactions on Image Processing* 15.2 (Feb. 2006), pp. 430–444. ISSN: 1941-0042. DOI: 10.1109/TIP.2005.859378. URL: https://ieeexplore.ieee.org/document/1576816 (visited on 12/11/2025).

EPFL