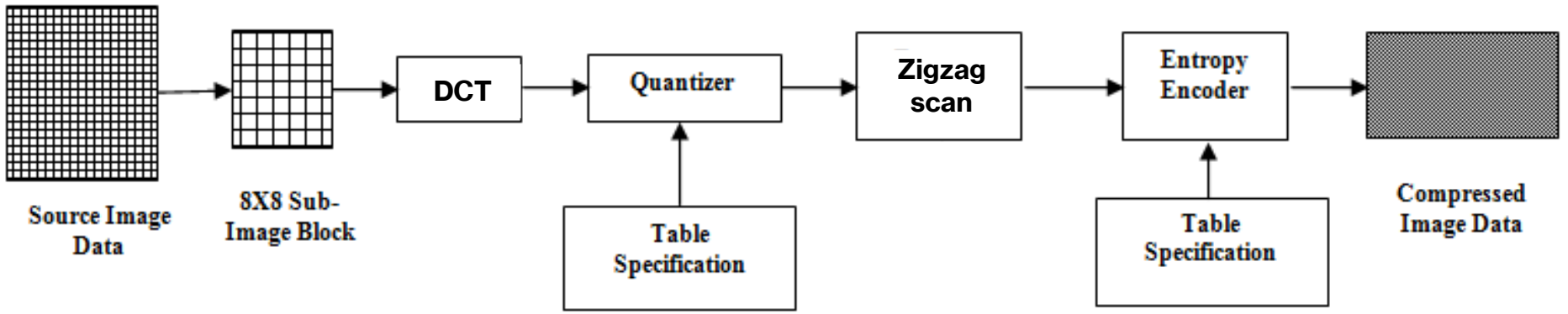
Visual Media Compression

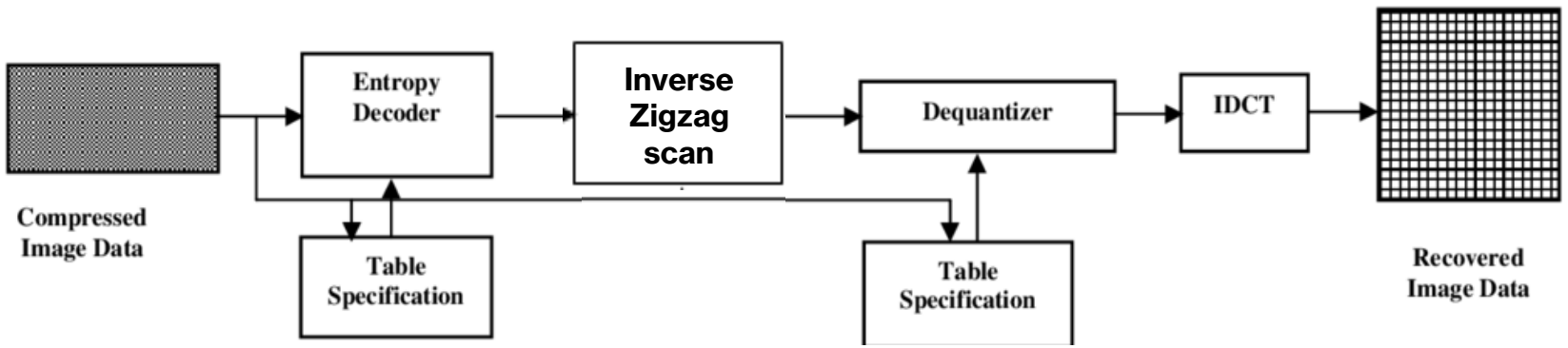
Yahya Bakkali

Image Codec

Encoder



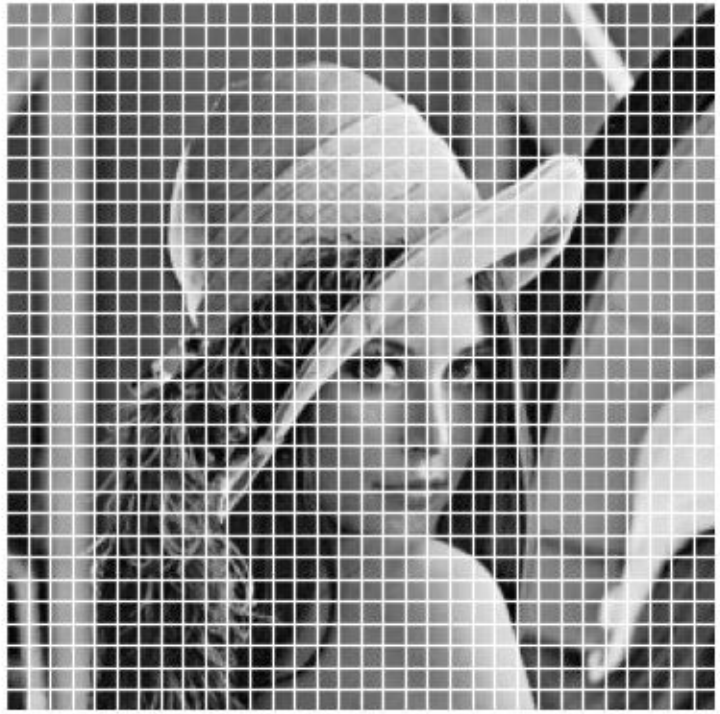
Decoder



Source : https://www.researchgate.net/publication/276198827_A_New_Hybrid_JPEG_Symbol_Reduction_Image_Compression_Technique

Encoder: Block-based coding

Source image



8x8 sub-image blocks

Encoder: DCT

DCT II

$$y_k = 2 \sum_{n=0}^{N-1} x_n \cos\left(\frac{\pi k(2n+1)}{2N}\right) \qquad \text{for } k = 0, \ldots, N-1.$$

Normalization factor

$$f = \begin{cases} \sqrt{\frac{1}{4N}} & \text{if } k = 0, \\ \sqrt{\frac{1}{2N}} & \text{otherwise} \end{cases}$$

DCT on Block 1

$$\begin{bmatrix} 137 & 136 & 133 & 136 & 138 & 134 & 134 & 132 \\ 137 & 136 & 133 & 136 & 138 & 134 & 134 & 132 \\ 138 & 133 & 134 & 134 & 136 & 132 & 130 & 130 \\ 133 & 133 & 133 & 130 & 134 & 133 & 128 & 125 \\ 129 & 133 & 130 & 130 & 133 & 131 & 132 & 128 \\ 131 & 133 & 130 & 122 & 132 & 131 & 130 & 130 \\ 131 & 130 & 130 & 130 & 132 & 131 & 128 & 130 \\ 131 & 132 & 130 & 130 & 131 & 131 & 130 & 128 \end{bmatrix} \xrightarrow{\text{DCT}} \begin{bmatrix} 1056 & 7 & -3 & 7 & 0 & -4 & -2 & 3 \\ 15 & 3 & -2 & 1 & 4 & 2 & -2 & 0 \\ 6 & -1 & 1 & -1 & 3 & 2 & -1 & -2 \\ -1 & -4 & 3 & 0 & -2 & -3 & -2 & 1 \\ -1 & 0 & -3 & 0 & -2 & 1 & -2 & -1 \\ -2 & 2 & -1 & 0 & 1 & 0 & 3 & 0 \\ -1 & 1 & 3 & 1 & -1 & -1 & 1 & 2 \\ 1 & -1 & -1 & -2 & 3 & 2 & -1 & -1 \end{bmatrix}$$

Source : https://docs.scipy.org/doc/scipy/reference/generated/scipy.fftpack.dct.html#scipy- fftpack- dct

Encoder: Quantization

Quantization matrix

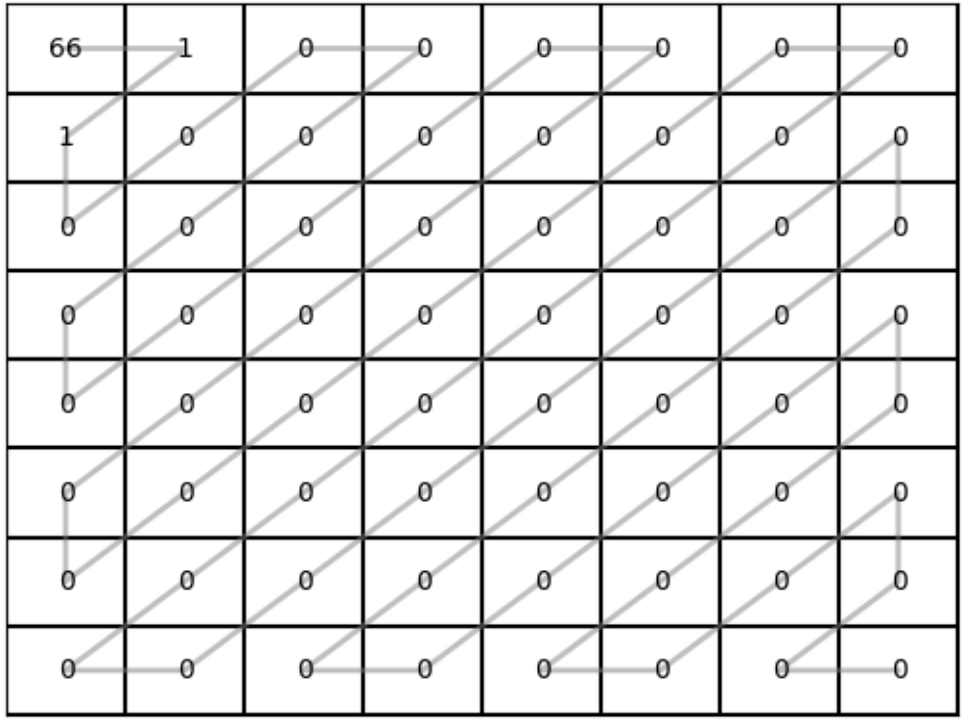
$$Q = \begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix}$$

Quantization on block 1

$$\begin{bmatrix} 1056 & 7 & -3 & 7 & 0 & -4 & -2 & 3 \\ 15 & 3 & -2 & 1 & 4 & 2 & -2 & 0 \\ 6 & -1 & 1 & -1 & 3 & 2 & -1 & -2 \\ -1 & -4 & 3 & 0 & -2 & -3 & -2 & 1 \\ -1 & 0 & -3 & 0 & -2 & 1 & -2 & -1 \\ -2 & 2 & -1 & 0 & 1 & 0 & 3 & 0 \\ -1 & 1 & 3 & 1 & -1 & -1 & 1 & 2 \\ 1 & -1 & -1 & -2 & 3 & 2 & -1 & -1 \end{bmatrix} \xrightarrow{/\ Q} \begin{bmatrix} 66 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Encoder: Zigzag scan

Zigzag pattern

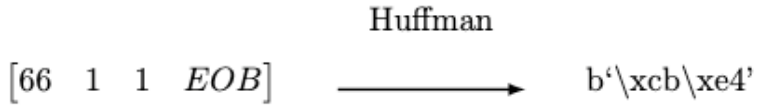


2D block 1 -> 1D

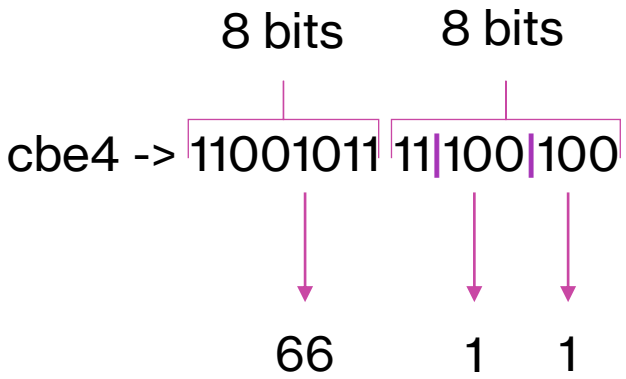
[66, 1, 1, EOB]

Encoder: Entropy Coding (Huffman)

Variable-length code on block 1



Hex -> binary



Decoder: Entropy Decoding & Inverse Zigzag scan

- Lossless
- Entropy decoding
  - Codebook
- Inverse Zigzag
  - 1D->2D
  - Add zeros

Decoder: Inverse Quantization

Dequantization of block 1

$$\begin{bmatrix} 66 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \xrightarrow{\times Q} \begin{bmatrix} 1056 & 11 & 0 & 0 & 0 & 0 & 0 & 0 \\ 12 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Lossy

$$\begin{bmatrix} 1056 & 7 & -3 & 7 & 0 & -4 & -2 & 3 \\ 15 & 3 & -2 & 1 & 4 & 2 & -2 & 0 \\ 6 & -1 & 1 & -1 & 3 & 2 & -1 & -2 \\ -1 & -4 & 3 & 0 & -2 & -3 & -2 & 1 \\ -1 & 0 & -3 & 0 & -2 & 1 & -2 & -1 \\ -2 & 2 & -1 & 0 & 1 & 0 & 3 & 0 \\ -1 & 1 & 3 & 1 & -1 & -1 & 1 & 2 \\ 1 & -1 & -1 & -2 & 3 & 2 & -1 & -1 \end{bmatrix} \neq \begin{bmatrix} 1056 & 11 & 0 & 0 & 0 & 0 & 0 & 0 \\ 12 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Decoder: Inverse DCT (IDCT)

IDCT II (DCT III)

$$y_k = x_0 + 2\sum_{n=1}^{N-1} x_n \cos\left(\frac{\pi(2k+1)n}{2N}\right) \qquad \text{for } k = 0, \ldots, N-1.$$

Normalization fomrula

$$y_k = \frac{x_0}{\sqrt{N}} + \sqrt{\frac{2}{N}} \sum_{n=1}^{N-1} x_n \cos\left(\frac{\pi(2k+1)n}{2N}\right) \text{ for } k = 0, \ldots, N-1.$$

IDCT on Block 1

$$\begin{bmatrix} 1056 & 11 & 0 & 0 & 0 & 0 & 0 & 0 \\ 12 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \xrightarrow{\text{IDCT}} \begin{bmatrix} 136 & 136 & 135 & 134 & 134 & 133 & 132 & 132 \\ 136 & 135 & 135 & 134 & 133 & 133 & 132 & 132 \\ 135 & 135 & 134 & 134 & 133 & 132 & 132 & 131 \\ 134 & 134 & 133 & 133 & 132 & 131 & 131 & 131 \\ 133 & 133 & 133 & 132 & 131 & 131 & 130 & 130 \\ 133 & 132 & 132 & 131 & 130 & 130 & 129 & 129 \\ 132 & 132 & 131 & 131 & 130 & 129 & 129 & 128 \\ 132 & 132 & 131 & 130 & 130 & 129 & 128 & 128 \end{bmatrix}$$

Source : https://docs.scipy.org/doc/scipy/reference/generated/scipy.fftpack.dct.html#scipy- fftpack- dct

Decoder: Blocks Merge

Decoded image 1 block

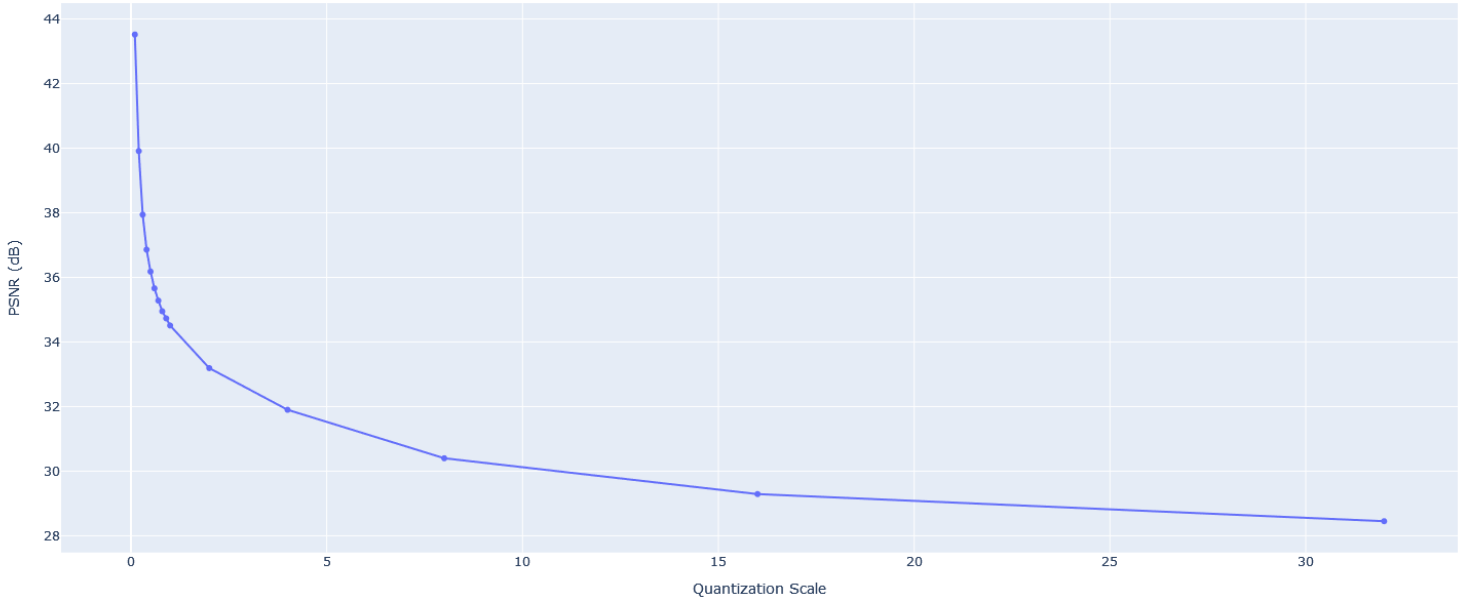
$$\begin{bmatrix} 137 & 136 & 133 & 136 & 138 & 134 & 134 & 132 \\ 137 & 136 & 133 & 136 & 138 & 134 & 134 & 132 \\ 138 & 133 & 134 & 134 & 136 & 132 & 130 & 130 \\ 133 & 133 & 133 & 130 & 134 & 133 & 128 & 125 \\ 129 & 133 & 130 & 130 & 133 & 131 & 132 & 128 \\ 131 & 133 & 130 & 122 & 132 & 131 & 130 & 130 \\ 131 & 130 & 130 & 130 & 132 & 131 & 128 & 130 \\ 131 & 132 & 130 & 130 & 131 & 131 & 130 & 128 \end{bmatrix} \neq \begin{bmatrix} 136 & 136 & 135 & 134 & 134 & 133 & 132 & 132 \\ 136 & 135 & 135 & 134 & 133 & 133 & 132 & 132 \\ 135 & 135 & 134 & 134 & 133 & 132 & 132 & 131 \\ 134 & 134 & 133 & 133 & 132 & 131 & 131 & 131 \\ 133 & 133 & 133 & 132 & 131 & 131 & 130 & 130 \\ 133 & 132 & 132 & 131 & 130 & 130 & 129 & 129 \\ 132 & 132 & 131 & 131 & 130 & 129 & 129 & 128 \\ 132 & 132 & 131 & 130 & 130 & 129 & 128 & 128 \end{bmatrix}$$

Original

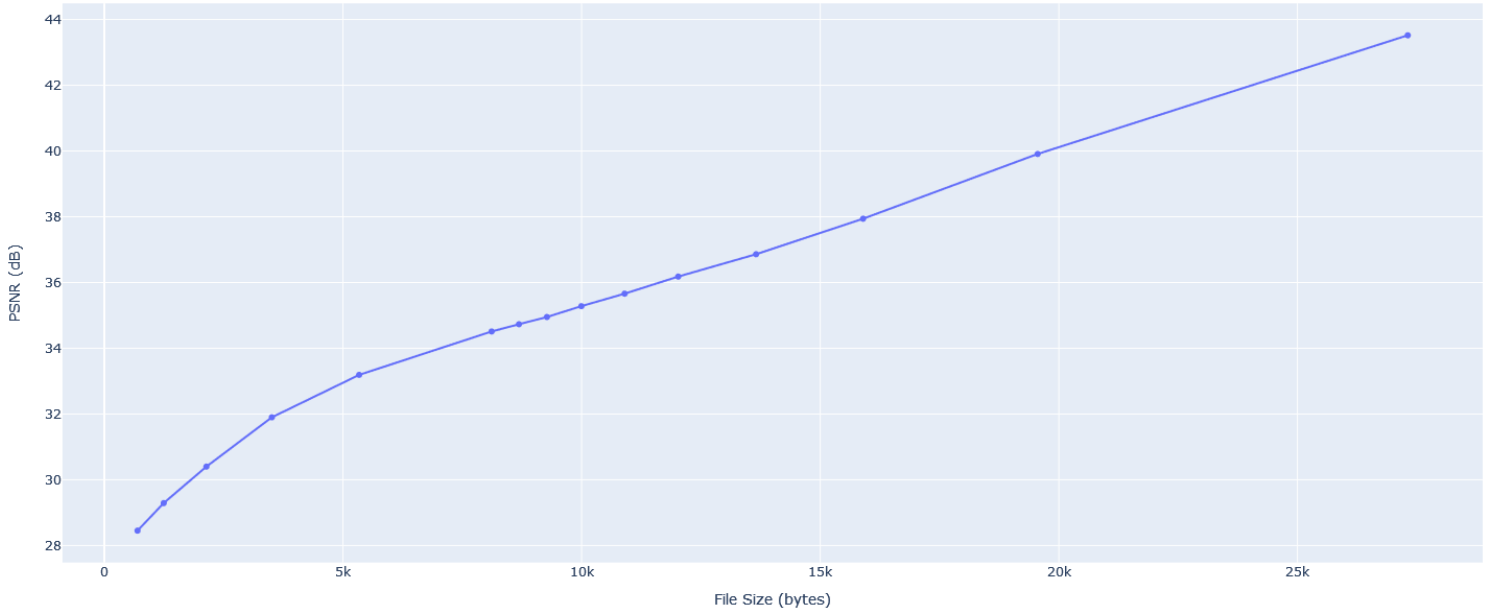
Compressed

Evaluation: Rate-Distortion Curve

PSNR vs. Quantization scale



PSNR vs. File Size



Evaluation: Subjective

- Difference undetectable by human eyes between a, b and c
- Good compression for 0.5 BPP

(a) 3.5 BPP

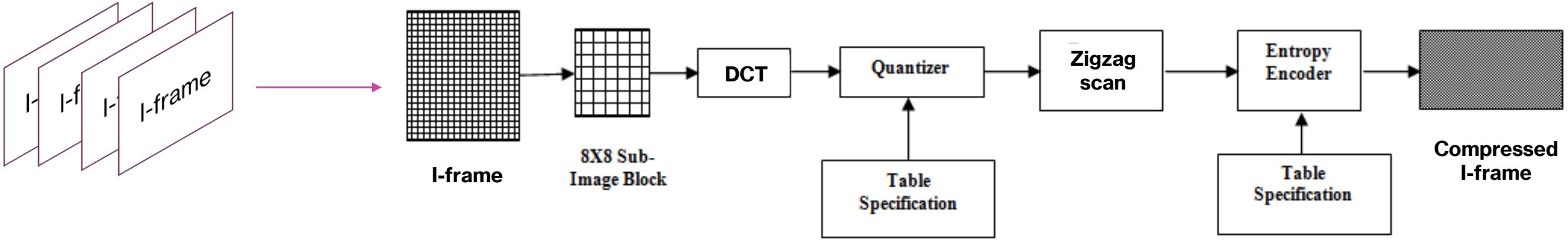
(b) 2 BPP

(c) 1 BPP

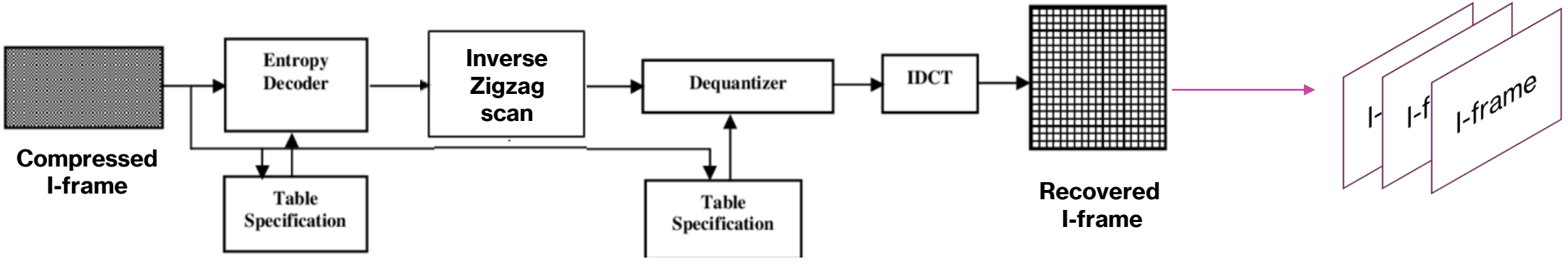
(d) 0.5 BPP

Video Codec (I-frames GOP)

Encoder

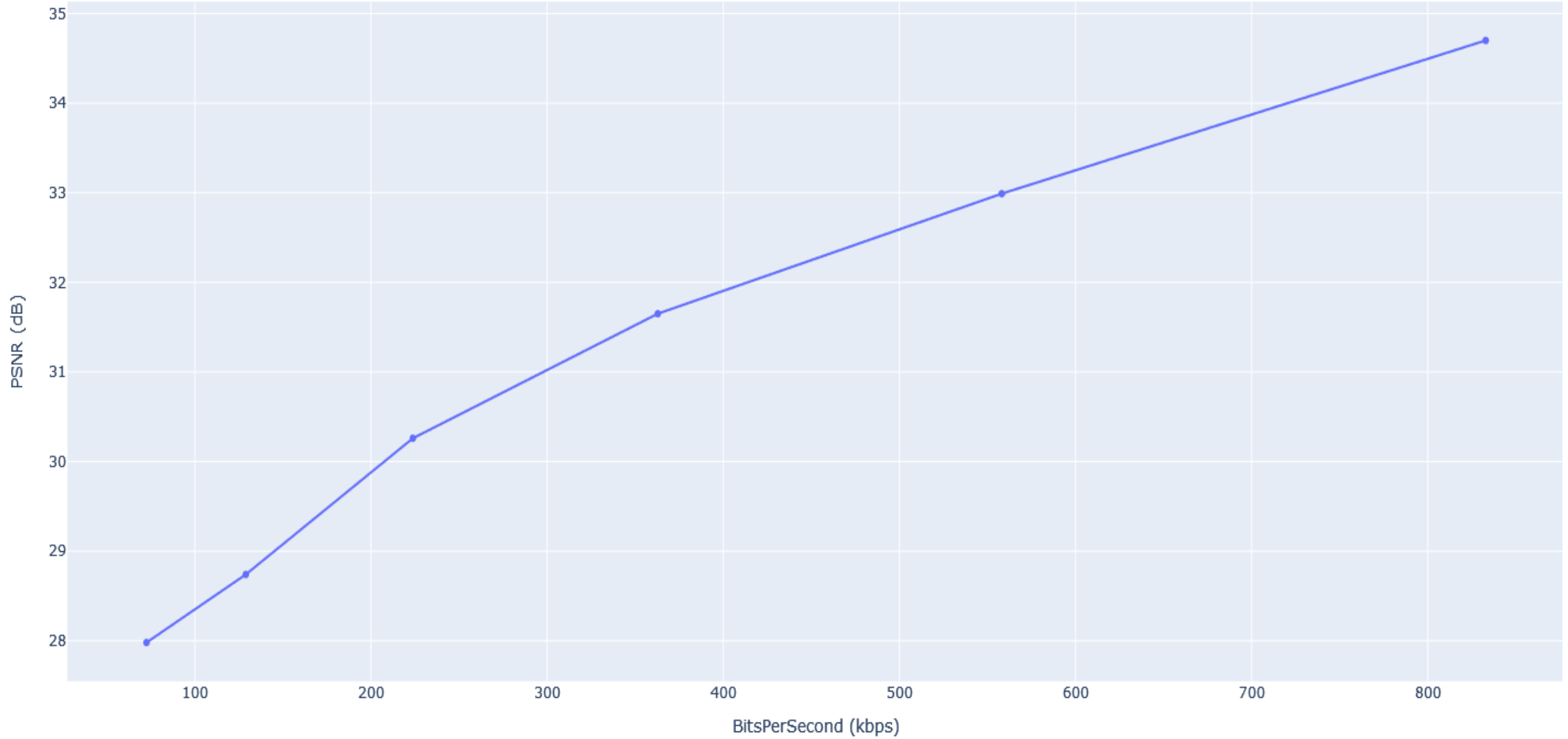


Decoder



Evaluation: Rate-Distortion Curve

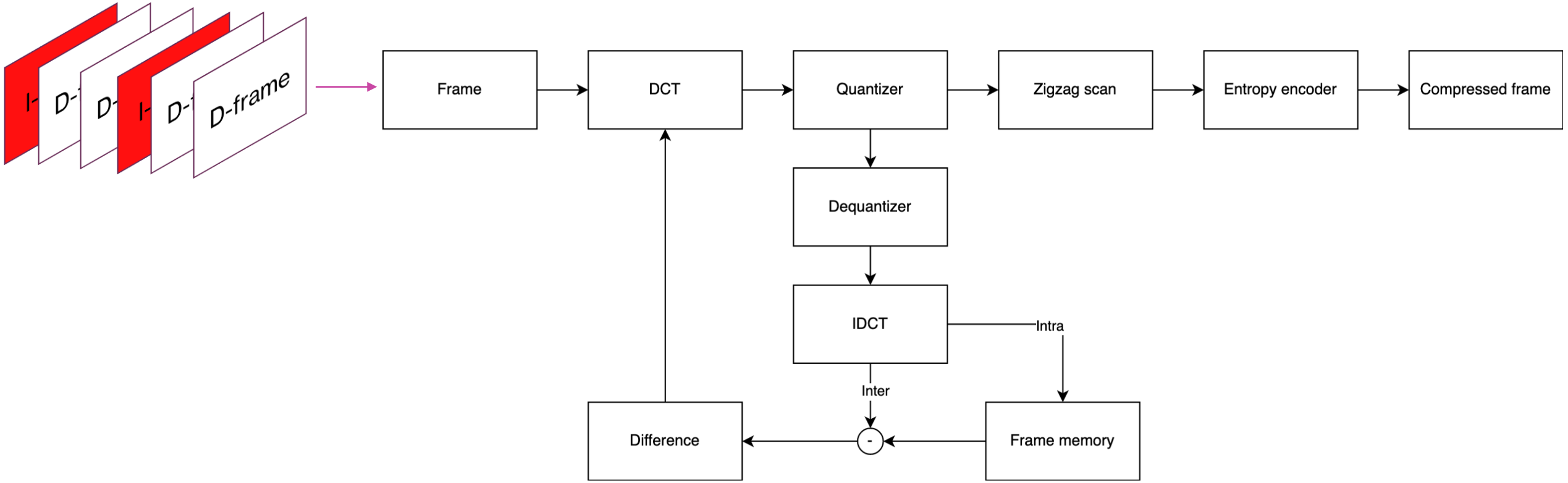
PSNR vs. BitsPerSecond



Evaluation: Subjective

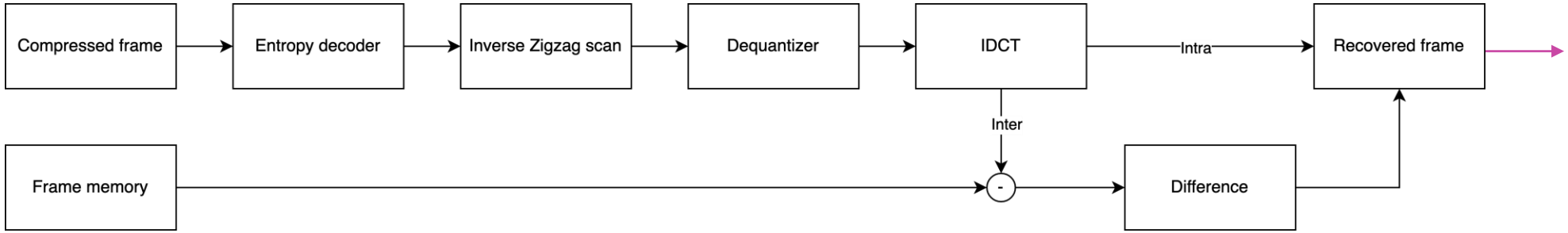
Video Codec (I-frames & D-frames GOP)

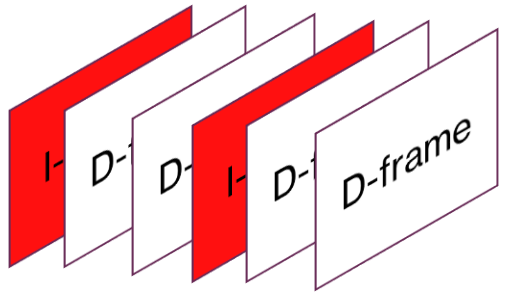
Encoder



Video Codec (I-frames & D-frames GOP)

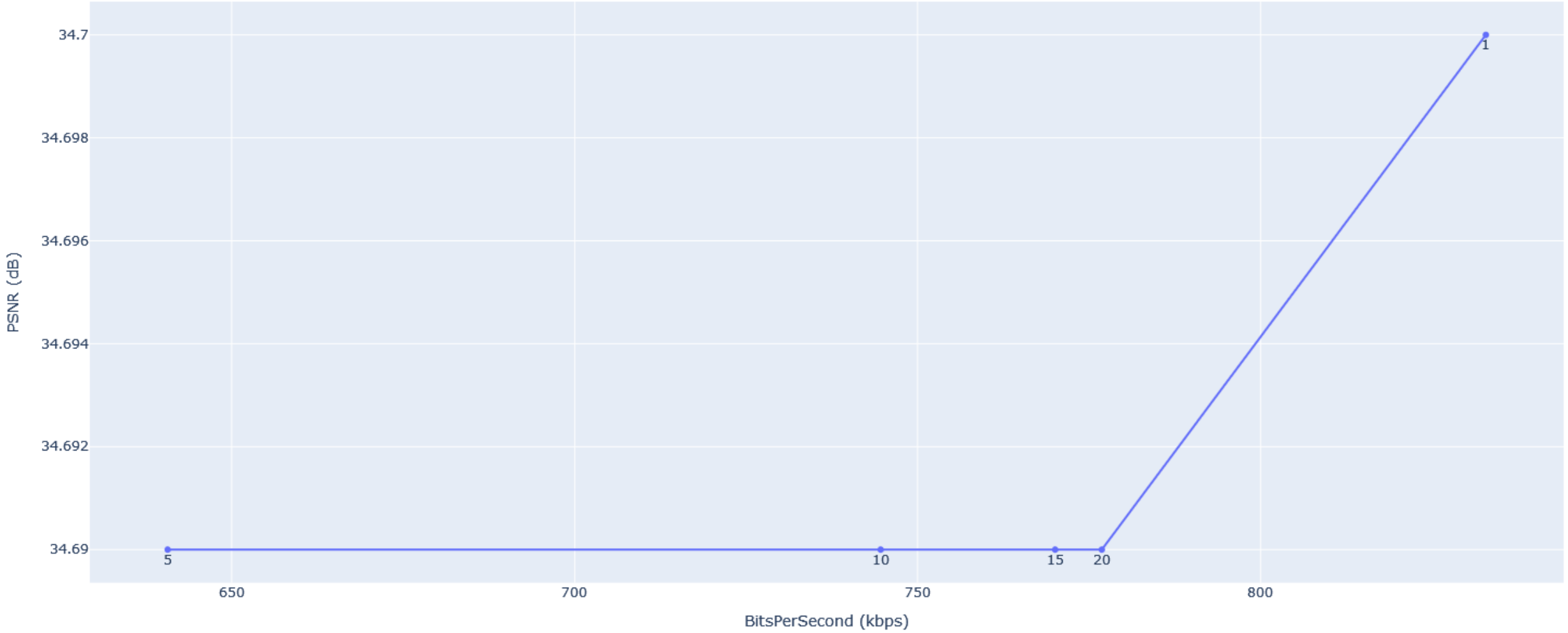
Decoder





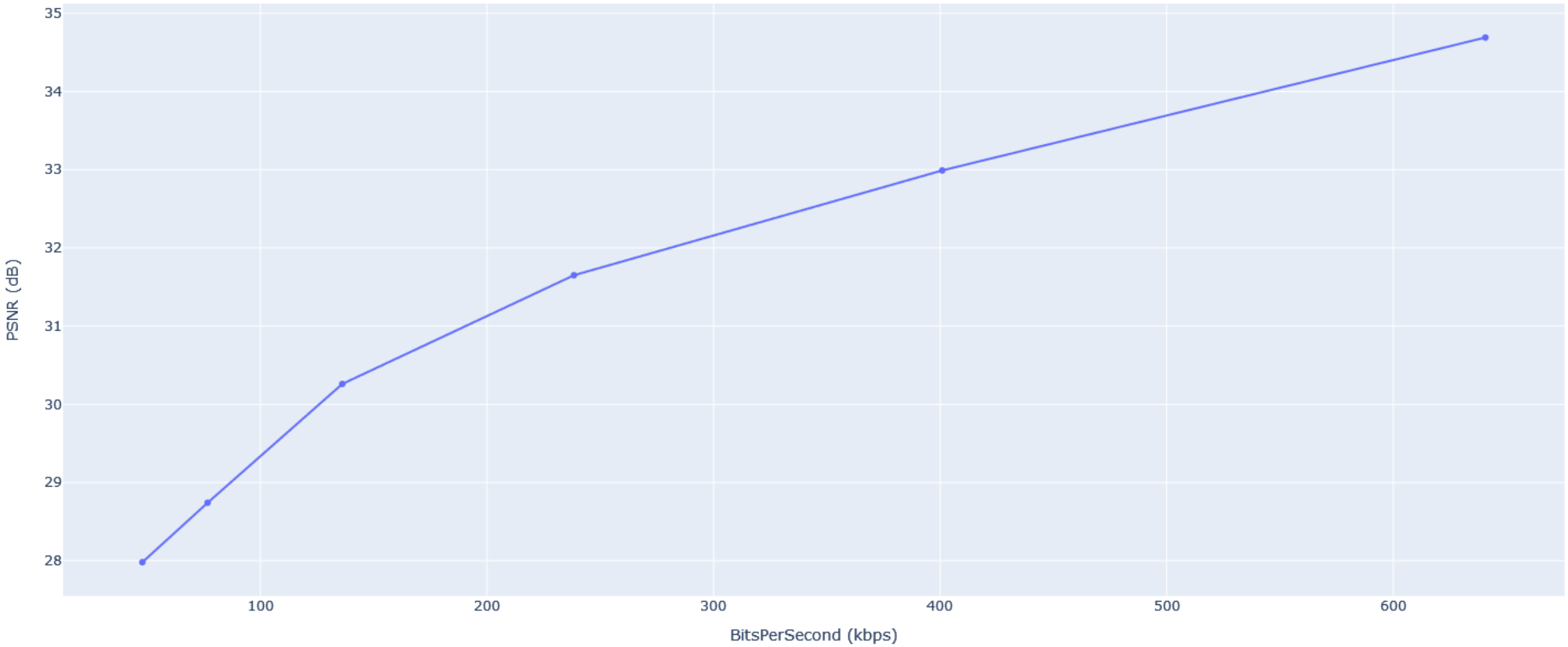
Evaluation: Rate-Distortion Curve

PSNR vs. BitsPerSecond for different GOP sizes



Evaluation: Rate-Distortion Curve of GOP 5

PSNR vs. BitsPerSecond



Evaluation: Subjective

Image Codecs Evaluation

PSNR vs. File Size

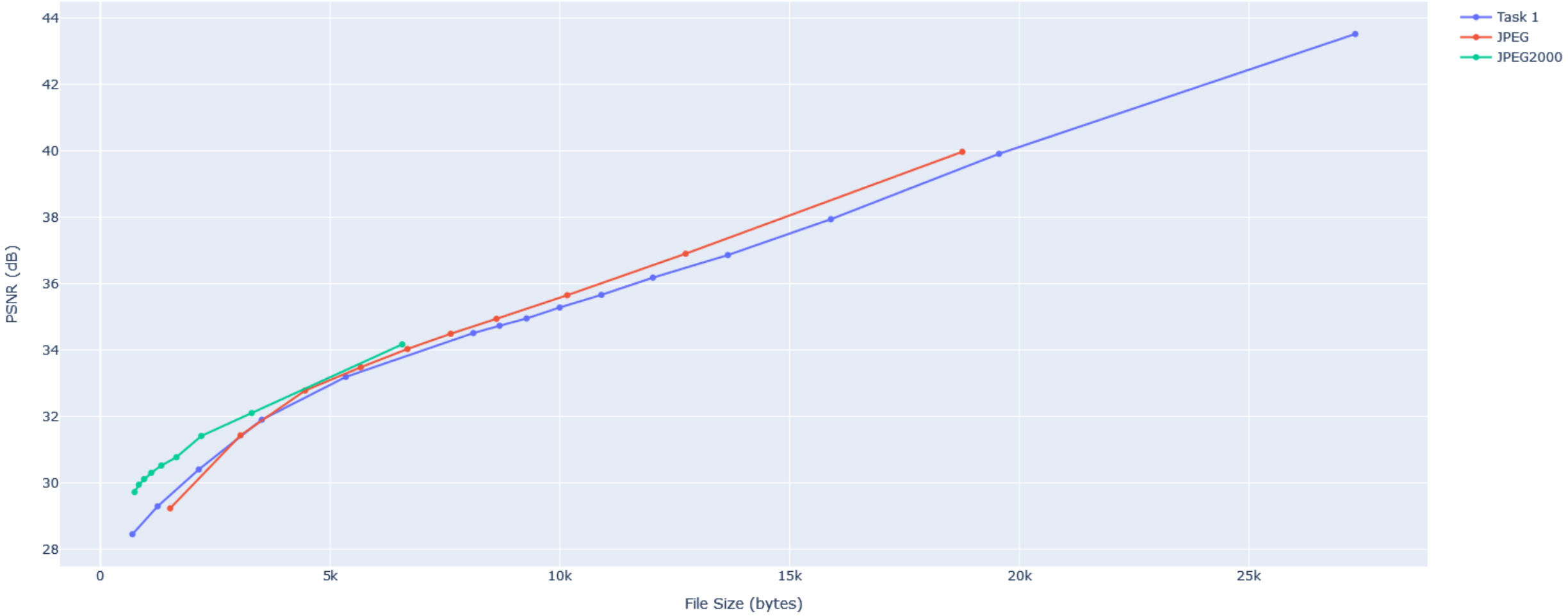


Image Codecs Evaluation

(a) Original (65.536 bytes)

(b) Task 1 (3055 bytes)

(c) JPEG (3048 bytes)

(d) JPEG2000 (2194 bytes)

Video Codecs Evaluation

PSNR vs. BitsPerSecond