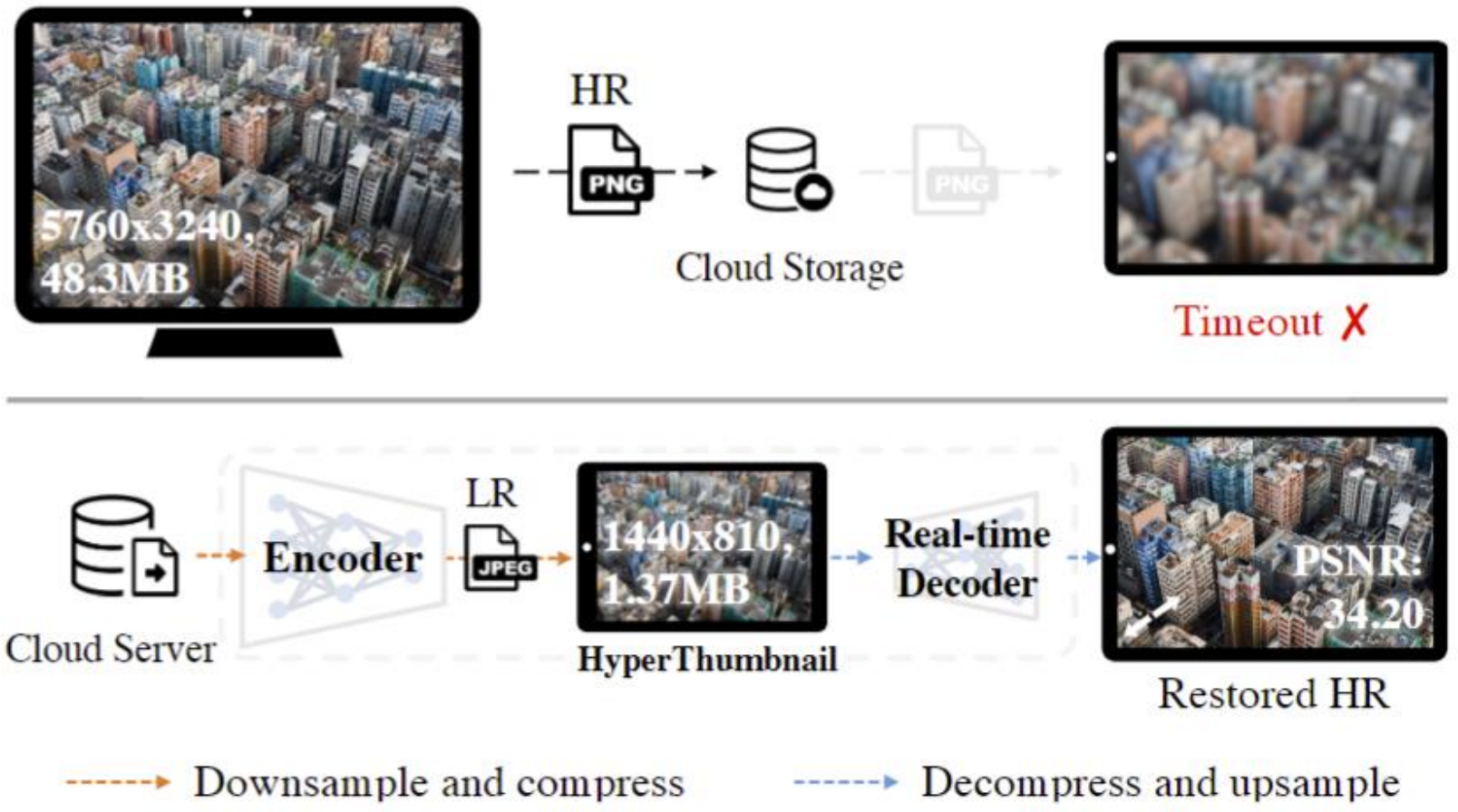
Real-time 6K Image Rescaling with Rate-distortion Optimization

Chenyang Qi*, Xin Yang*, Ka Leong Cheng, Yingcong Chen, Qifeng Chen

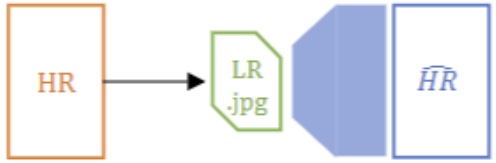
Project link: https://github.com/AbnerVictor/HyperThumbnail

*: Equal contribution

Motivation of Our Task



Previous framework

Method	(a) Downsampled JPEG + super-resolution [36]	(b) Flow-based rescaling [35, 54]	(c) Ours
Architecture			
Reconstruction fidelity	✗	✓	✓
Rate-distortion optimization	✗	✗	✓
Real-time 6K reconstruction	–	✗	✓

Our performance contributions

1. Best Bitrate-Distortion
   - Previous: even worse than JPEG
   - Ours: comparable with learned compression

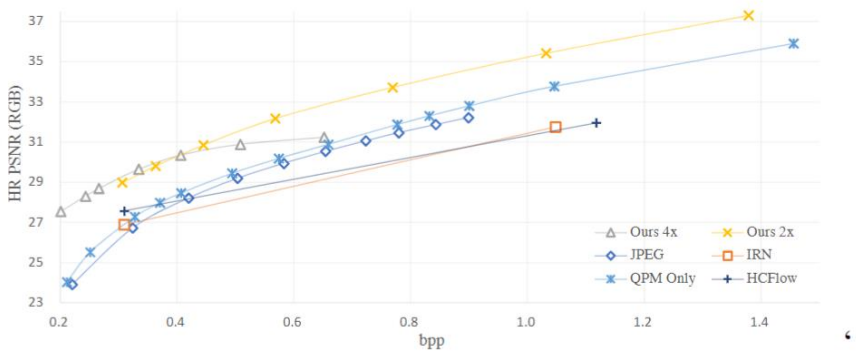
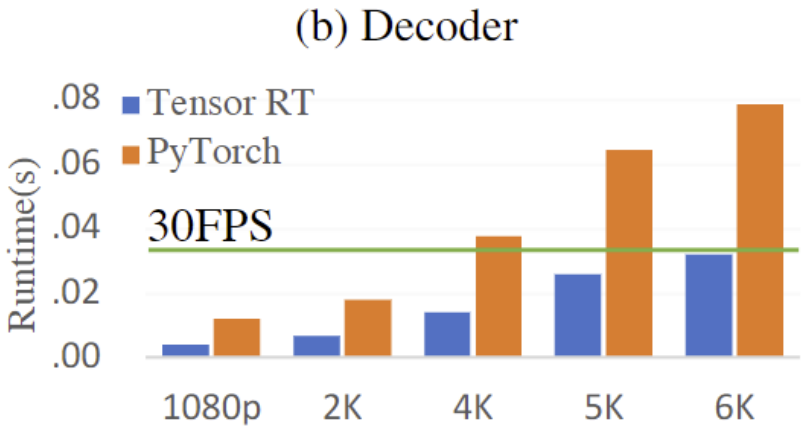
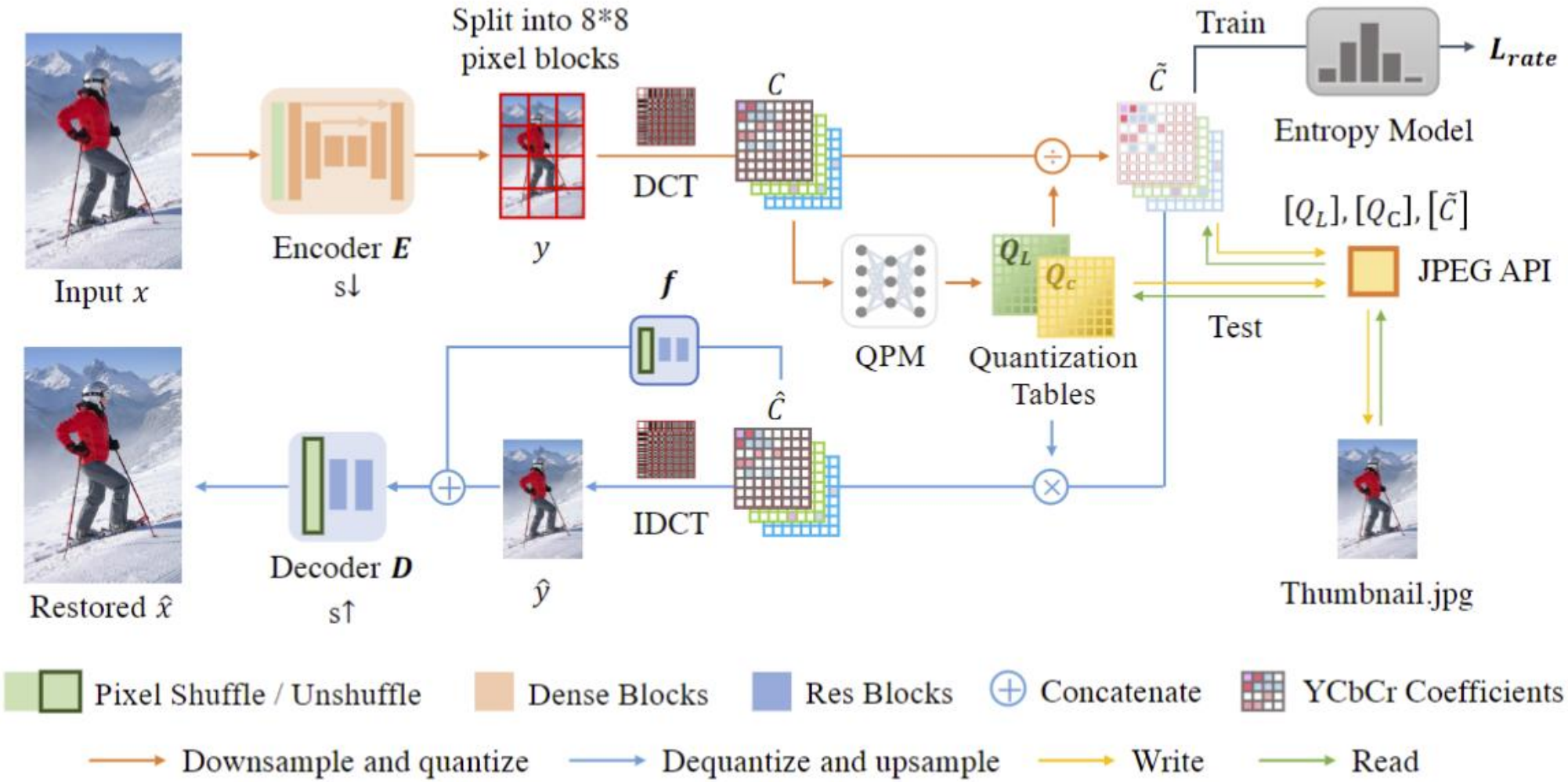


Figure 1. **The rate-HR-distortion curve on Kodak [1] dataset.** Our method ($s = 2, 4$) outperforms JPEG, IRN [13] and HCFlow [8] in the RD performance. Especially, for the 'QPM Only' curve, where $s = 1$, we follow standard JPEG algorithm and adopt QPM module as a plugin for table prediction.

2. Efficient Upscaling time
   - Previous: > 1s for 4K image reconstruction
   - Ours: 6K real-time under TensorRT



Our Architecture

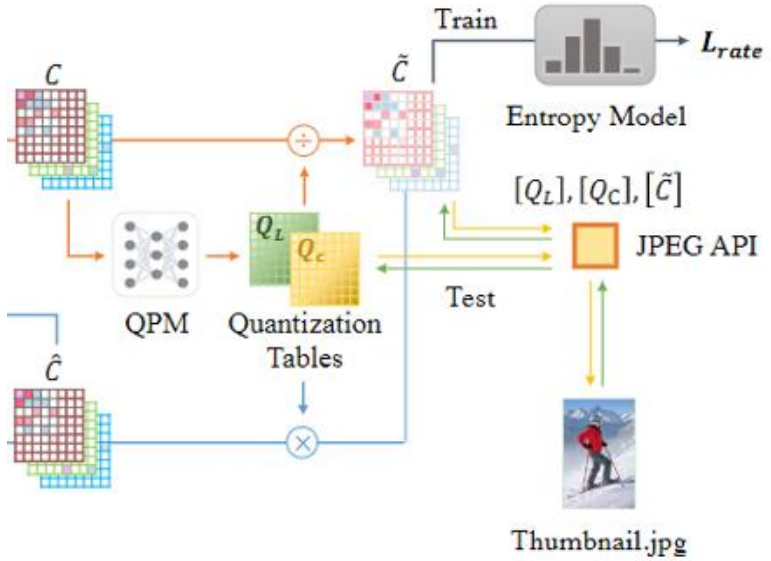


Rate-Distortion objective during training

Rate:

$$R = \mathbb{E}_{x \sim p_x}[-log_2\, p_L(\widetilde{C}_Y) - log_2\, p_C(\widetilde{C}_{Cb}) - log_2\, p_C(\widetilde{C}_{Cr})],$$

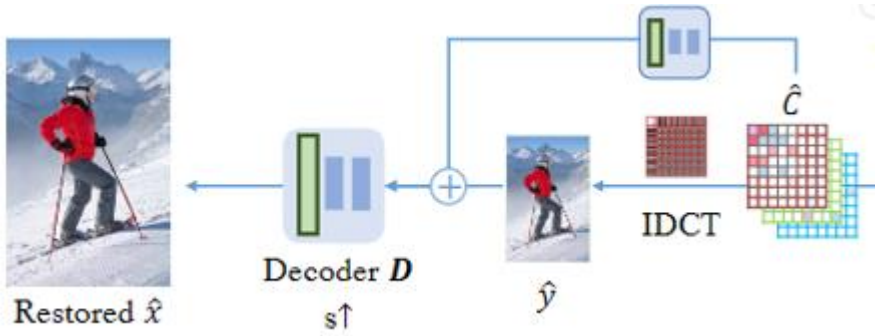
$$L_{bpp} = \frac{R}{H \times W}.$$



Distortion:

$$L_{recon} = \frac{||\hat{x} - x||_1}{H \times W},$$

$$L_{guide} = \frac{||\hat{y} - y_{ref}||_2^2}{(H/s) \times (W/s)},$$



Total:

$$L_{rescale} = L_{recon} + \lambda_1 L_{guide} + \lambda_2 L_{bpp}.$$

Quantitative result: Bitrate-Distortion-Efficiency

Method	Bitrate↓-Distortion↑ [1]		Upscaling Efficiency↓		Reconstructed HR PSNR↑					
Down & Degradation & Up	bpp	PSNR	Time (ms)	GMacs	Set5	Set14	BSD100	Urban100	DIV2K	FiveK-6k
Bicubic & JPEG & Bicubic	0.29	25.18	–	–	25.14	23.49	24.02	21.05	25.70	26.90
Bicubic & JPEG & EDSR [36]	0.29	26.77	91.0	1007.5	28.34	25.73	25.60	23.58	27.83	27.23
Bicubic & JPEG & SwinIR [34]	0.29	26.93	4012.6	6208.7	28.56	25.99	25.72	24.09	28.07	27.44
ComCNN & RecCNN [26]	0.32	27.02	469.7	6014.7	28.29	25.84	25.78	23.70	27.99	27.40
IRN [54] & JPEG	0.31	28.48	977.8	4751.7	30.00	27.23	26.91	25.72	29.54	27.96
HCFlow [35] & JPEG	0.30	28.76	1025.9	4626.0	29.98	27.41	27.05	26.19	29.71	28.01
Ours-full	0.30	**29.67**	247.9	1277.5	**30.48**	**28.21**	**27.93**	**27.35**	**30.49**	**28.51**
Ours	0.30	29.42	**37.8**	**156.2**	30.22	27.87	27.66	26.62	30.15	28.15

Qualitative result of reconstructed HR

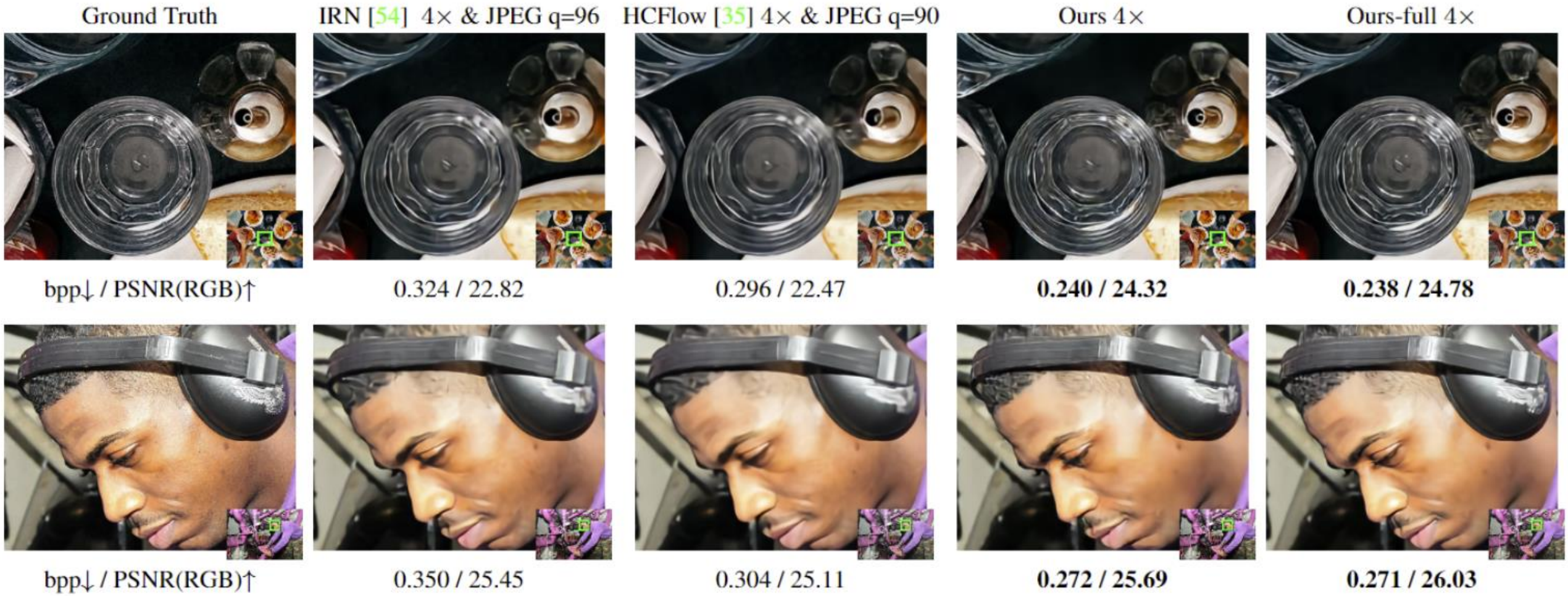


Figure 3. **Reconstructed HR images and LR thumbnails by different methods on the DIV2K [4] validation dataset.** We crop the restored HR images to ease the comparison and visualize the LR counterparts at the bottom-right. The bpp is calculated on the whole image and the PSNR is evaluated on the cropped area of the reconstructed HR images.

Evaluation of LR thumbnail

Method	LR thumbnail PSNR↑						
Down & Degradation	Kodak	Set5	Set14	BSD100	Urb100	DIV2K	FiveK-6k
Bicubic & JPEG	37.72	34.41	35.79	37.43	36.64	37.14	35.14
IRN [54] & JPEG	30.95	30.00	27.23	26.91	25.72	29.54	31.25
ComCNN & RecCNN [26]	28.00	26.76	26.47	27.47	25.57	28.15	28.99
HCFlow [35] & JPEG	19.88	20.08	19.42	19.65	18.96	20.52	20.31
Ours-full	33.21	31.86	31.76	32.44	31.01	33.32	33.99
Ours	33.55	31.96	31.93	32.90	31.16	33.62	34.24

Table 3. Quantitative evaluation of the 4× downsampled LR thumbnails by different methods. The target bitrate is around 0.3 bpp on Kodak [1] for all methods, and we take Bicubic LR as the ground truth. Our thumbnail preserves visual contents better.

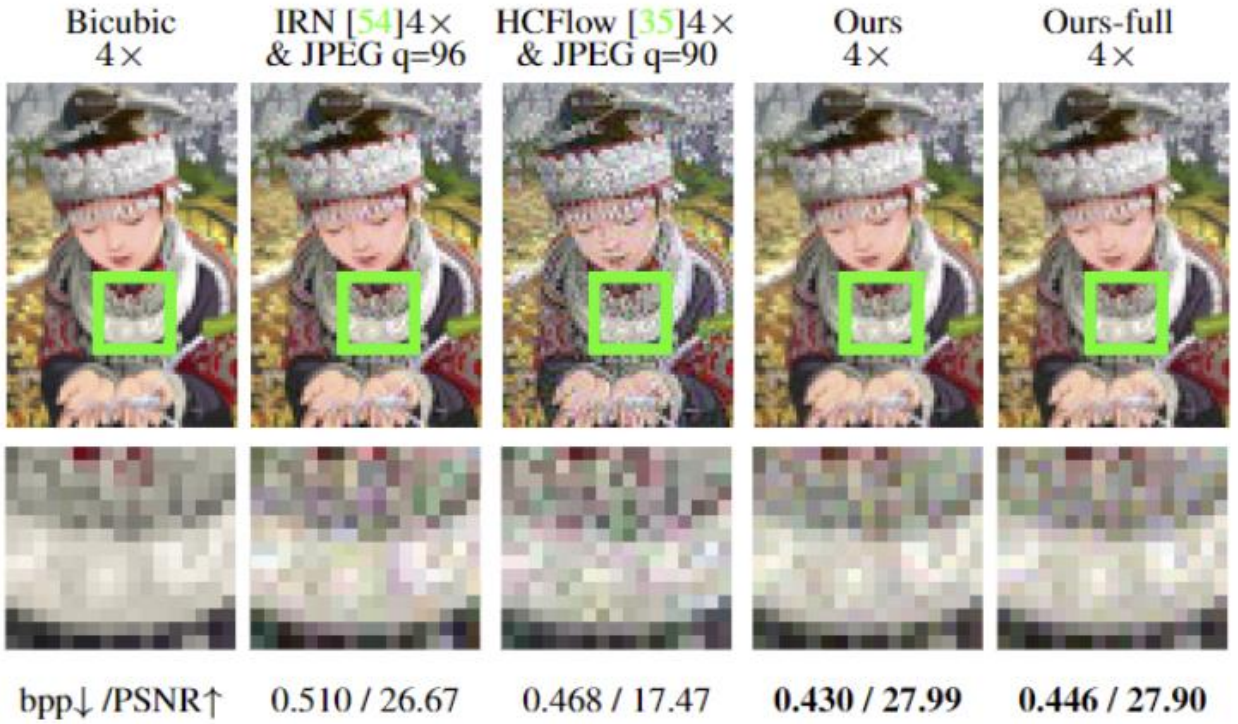


Figure 4. **Downscaled LR thumbnails by different methods on Set14 image *comic*.** With a similar target bpp, our model introduces least artifacts in the thumbnail in comparison to baselines.

Compare Bit-distortion with other methods

Compare with rescaling use PNG

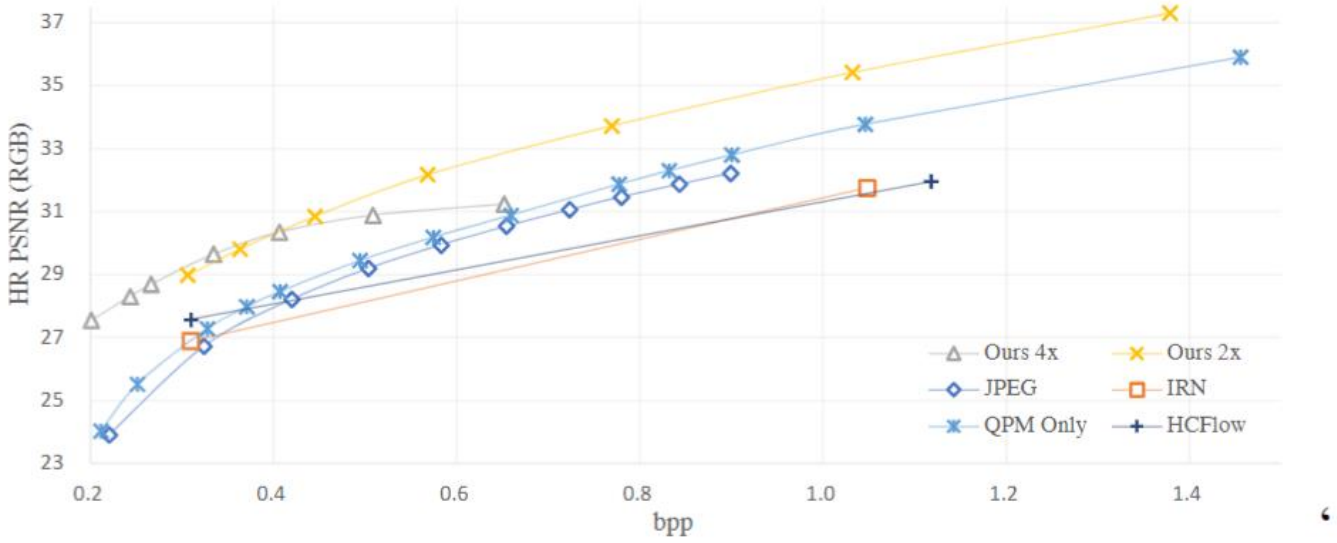


Figure 1. **The rate-HR-distortion curve on Kodak [1] dataset.** Our method ($s = 2, 4$) outperforms JPEG, IRN [13] and HCFlow [8] in the RD performance. Especially, for the 'QPM Only' curve, where $s = 1$, we follow standard JPEG algorithm and adopt QPM module as a plugin for table prediction.

Compare with compression with thumbnail

Method	Bpp of File Format		Bitrate↓-Distortion↑ Kodak	
Architecture	Bitstream	JPEG	Sum of bpp	LR PSNR / HR PSNR
Hyperprior [7]+JPEG	0.214	0.148	0.51	33.41 / 29.22
HIFIC [39]+JPEG	0.172	0.148	0.32	33.41 / 29.35
Ours	-	0.299	**0.30**	**33.55 / 29.42**

Table 7. Comparison of our HyperThumbnail framework against learned compression with JPEG thumbnail. In additional baseline, we provide a JPEG thumbnail besides learned compression, and take the sum of bitstream size and JPEG size to calculate the final bpp. Our framework has better rate-distortion performance than "Compression+JPEG" baseline.

Achieve 6K real-time Reconstruction

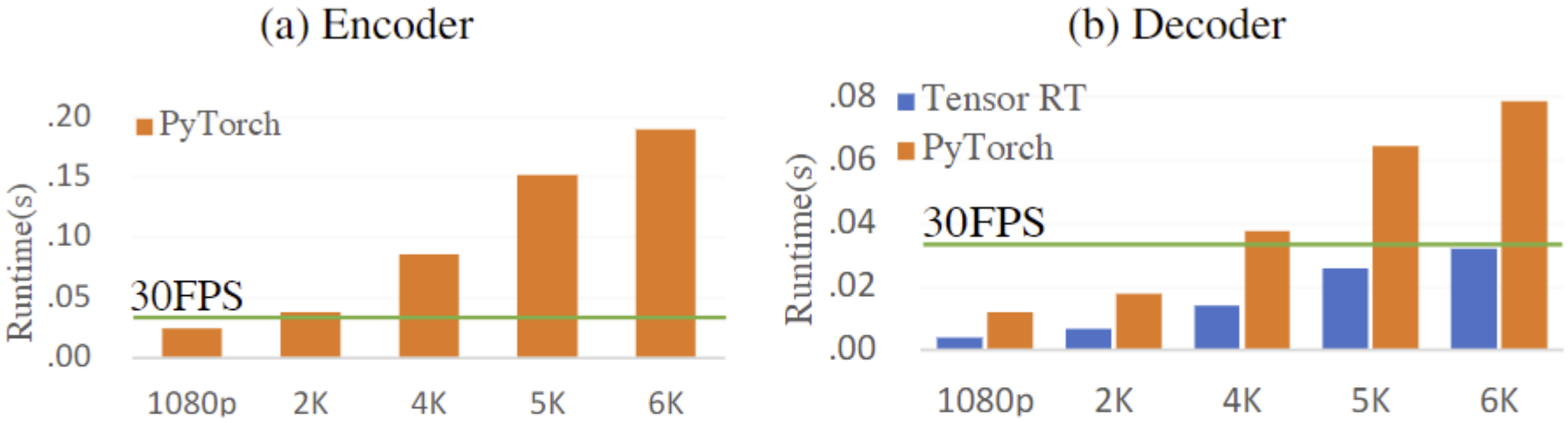


Figure 5. **Model runtime.** We profile the $4\times$ encoder and decoder at different target resolution in half-precision mode. Especially, we convert our decoder from PyTorch to TensorRT for further inference time reduction.

Ablation of Quantization Prediction Module

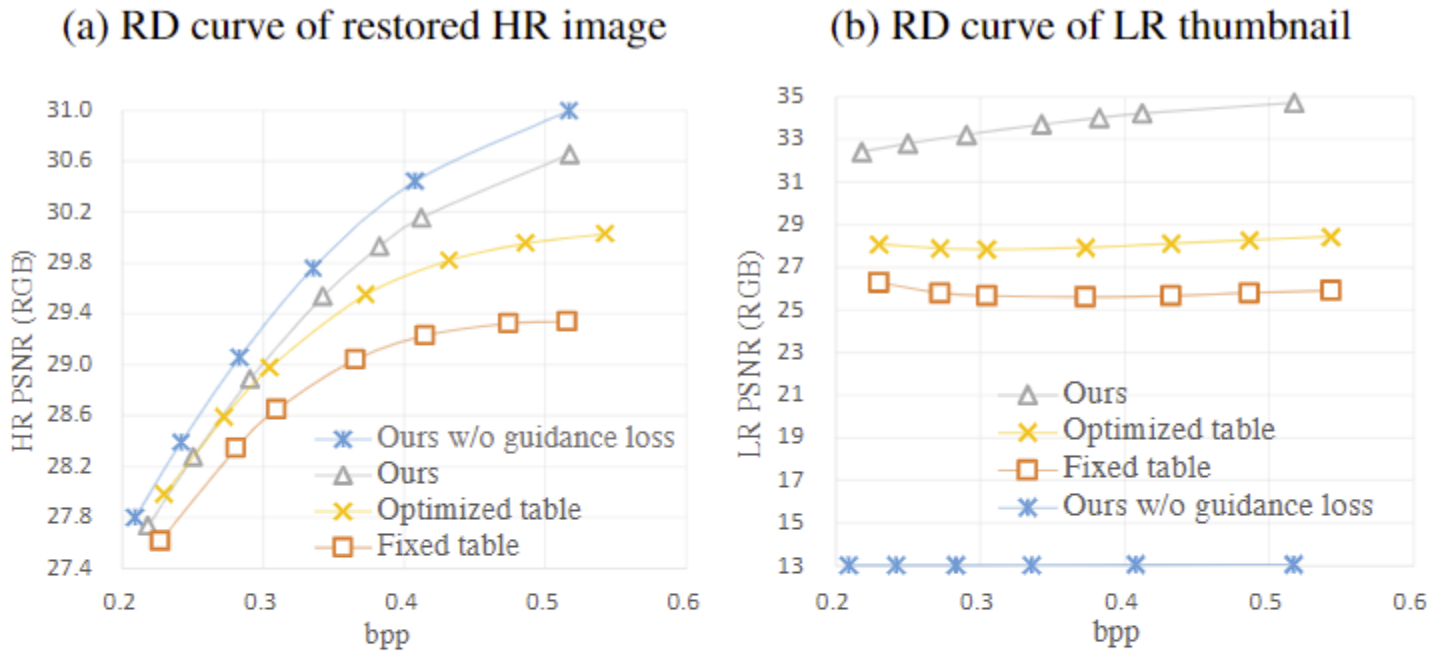
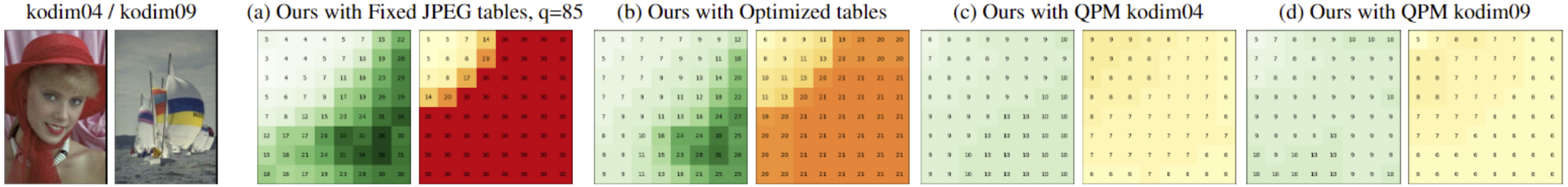


Figure 7. **QPM versus image-invariant quantization.** We first train our models with QPM, with a fixed JPEG table or with an optimized table, respectively. Then, we evaluate the at different target bitrate on Kodak [1] dataset. **(a)** the RD curve on reconstructed HR image $\hat{x}$ and input x; **(b)** the RD curve on LR thumbnail $\hat{y}$ and the Bicubic downsampled LR y_{ref}.

Analysis and Extension



bpp↓ / LR↑ / HR↑: *kodim09*	0.263 / 26.91 / 30.54	0.289 / 29.10 / 30.93	-	**0.246 / 31.03 / 34.34**
bpp↓ / LR↑ / HR↑: *kodim04*	0.251 / 26.96 / 31.35	0.276 / 29.17 / 31.71	**0.240 / 34.38 / 32.03**	-

Figure 6. **Quantization tables on Kodak [1] images.** We visualize the quantization table Q_L (the green table) and Q_C (the orange table) for *kodim04* and *kodim09* of different quantization approaches. The model trained with QPM achieves the best RD performance from every aspect. For more analysis, please refer to Sec. 5 in our paper.

Method	Optimization	bpp↓ / HR PSNR↑		
Architecture	iteration	Kodak	Set5	Set14
Ours	0	0.301 / 29.42	0.379 / 30.23	0.359 / 27.74
	100	0.307 / 29.55	0.377 / 30.36	0.347 / 27.96

Table 4. Quantitative evaluation for optimization-based rescaling.

Takeaways

- The first rescaling method to achieve 6K real-time decoding and sota rate-distortion performance.
- Introduce an entropy model for modeling jpeg image file size
- Demonstrate the importance of Rate-Distortion loss for practical rescaling

Thank you!