

Supplementary Material

A Additional Qualitative Results

Figs. 12 - 18 show additional qualitative comparisons between our method and the baselines from the main text.

We also show full image comparisons between the diffusion-based image compression baselines [25, 46] and strong autoencoder-based [21] and traditional [1] image codec baselines in Figs. 19 - 23.

B Additional Implementation Details

Foundation Model Backbone. We take Stable Diffusion v2.1 as our foundation model backbone, using the official code repository⁷ and model checkpoint⁸. We use the default configuration, except for minor modifications detailed below.

We utilize the DDIM sampling formulation for the diffusion generative process. As mentioned in Sec. 4.4, it is prohibitively expensive to accumulate gradients over multiple DDIM steps during training of our method, therefore we slightly modify the sampling procedure as follows. At each DDIM step, a prediction of the fully denoised sample $\hat{\mathbf{x}}_0$ is also computed (see Eq. (2)). Thus instead of performing multiple steps of the diffusion process to get the final sample, we can directly utilize $\hat{\mathbf{x}}_0$. Although $\hat{\mathbf{x}}_0$ is slightly inconsistent with the true fully denoised image, at the low timestep range our method operates in we observe that the difference is minor (see Fig. 8) and has negligible effect on optimization.

Stable Diffusion offers multiple prediction parameterizations, such ϵ or v prediction, which dictates the output of the diffusion model. We utilize ϵ prediction due to our modifications of the sampling algorithm during training. While the commonly used v prediction is more stable over the full generation process compared to ϵ prediction, in the low timestep range where our model operates v prediction reduces almost completely to ϵ prediction [39] and thus is effectively equivalent.

Entropy Model. Entroformer [37] is used as the entropy model for our method. We refer readers to the original paper and codebase⁹ for specific information. We follow the architectural details in Appendix A.4 of the original paper, but replace the encoder and decoder with the Stable Diffusion VAE encoder and decoder. Correspondingly, we also change the output dimension of the entropy model to match the channel dimension of the Stable Diffusion latent space (*i.e.* Entroformer configuration parameter `last_channels` = 4).

⁷ <https://github.com/Stability-AI/stablediffusion>

⁸ https://huggingface.co/stabilityai/stable-diffusion-2-1/blob/main/v2-1_768-ema-pruned.ckpt

⁹ <https://github.com/damo-cv/entroformer>



Fig. 8: Comparison of the predicted original sample during training (single diffusion step), at inference time (multiple diffusion steps), and the difference between the two (shown from left to right, respectively). Best viewed digitally.

C Similarity of Quantization Error and Noise

In signal processing, quantization error has historically been modeled as uniform [12] noise. Ballé *et al.* [7] first introduced this to the field of neural compression and since then it has been widely adopted in other neural image compression methods [8, 21, 34, 37].

Intuitively, given a quantization bin with lower and upper bounds a and b , respectively, all values within this range are mapped to the bin center $c = \frac{(b+a)}{2}$ during quantization. Thus, assuming a smooth distribution of values before quantization (as is the case in the latent space of a variational autoencoder), the error for values within one bin is approximately uniformly distributed across the bin width (*i.e.* $\mathcal{U}(a - c, b - c)$). Given a constant bin width, the error across all quantization bins is therefore also uniformly distributed.

D Details on User Study

Further Evaluation. In Sec. 5.1 of the main text we report median Elo across all participants and all images, and across all participants (*i.e.* we update Elo ratings of the methods after each game and after all comparisons of each participant, respectively). In Fig. 9 we additionally report Elo scores where all comparisons for each image are treated as a mini-tournament (*i.e.* we update Elo ratings of the methods after all comparisons of the same image). We report median Elo score over 10,000 Monte Carlo iterations as in the main text.

The overall ranking order of the methods does not change, although the variance of the Monte Carlo simulation is larger. This is because the Elo score update when a higher ranked method beats a lower ranked one is smaller, and as there are less frequent, aggregated score updates the “stronger” methods are not penalized due to their strength as much as when updating after every comparison.

Images Used. For all images used in our user study, we report the filenames and the bitrate (in bits per pixel) for each method in Tab. 1. We compress and decompress the entire image with each method and center crop the result to 512×512 px to display to the user (see Sec. 5).

User Interface. Fig. 10 shows a screenshot of the user interface of our user study. The original image is shown in the center, while left and right show the reconstructed images. The order in which each pair comparison appears and the position of each method (left/right) are randomly selected for each rating. Our user interface supports synchronized zooming and panning, so the participant can examine smaller areas of each image if preferred. Zoom and pan levels are reset for each new comparison.

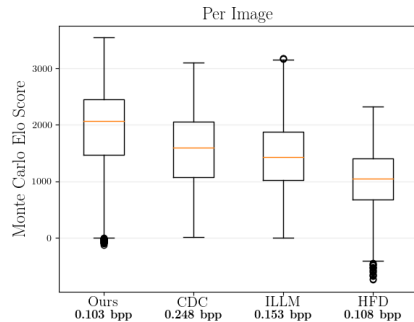


Fig. 9: Per-image Monte Carlo Elo ratings from the user study. Higher is better. The box extends to the first and third quartiles and the whiskers $1.5 \times IQR$ further.

	kodim01	kodim05	kodim07	kodim09	kodim08	kodim17	kodim19	kodim22	kodim23	kodim24
Ours	0.0852	0.154	0.101	0.0744	0.163	0.0927	0.0907	0.0984	0.0606	0.110
HFD	0.0940	0.161	0.111	0.0765	0.167	0.0966	0.0918	0.0992	0.0644	0.113
CDC	0.276	0.365	0.229	0.166	0.297	0.237	0.207	0.252	0.160	0.290
ILLM	0.0983	0.240	0.147	0.118	0.243	0.140	0.146	0.146	0.0644	0.187

Table 1: Filenames and corresponding bitrate of each method for all images used in the user study. Bitrate is expressed in bits per pixel.

E Further Analysis on Timestep Prediction

Here we present additional evidence that our method is able to predict the ideal number of denoising timesteps over all bitrates (see Sec. 4).

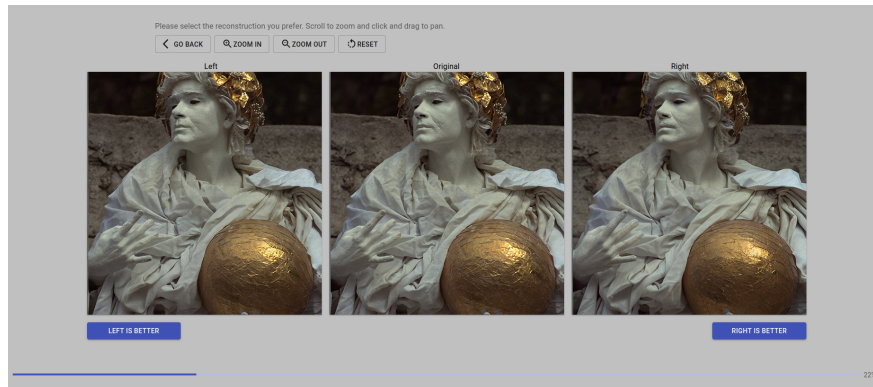


Fig. 10: Screenshot of the user interface of our subjective user study.

We perform an experiment in which we use our method to compress an image at multiple bitrates and manually sweep over a range of denoising diffusion timesteps (similar to the naive latent diffusion implementation in Sec. 3 of the main text, but only over the timestep parameter). We record rate-distortion metrics for each quantization level and timestep pair. Fig. 11 shows the results of this experiment on images from the Kodak dataset, where for all bitrates our method predicts the number of denoising diffusion steps which results in the lowest distortion.

F Video

We include a short video detailing an overview of our approach and additional visual comparisons in the supplementary material.

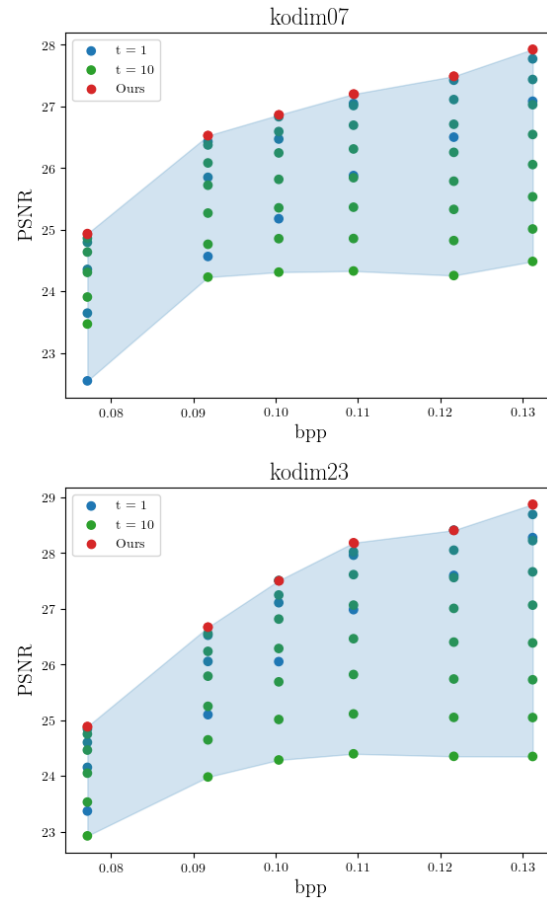


Fig. 11: Rate-distortion curve of our method with manually chosen denoising diffusion steps. The color gradient of the dots represents the number of denoising steps. Our predicted optimal number of steps is shown in red.



Fig. 12: Additional visual comparisons (kodim01). Images are labeled with Method@bpp (% bpp compared to Ours).

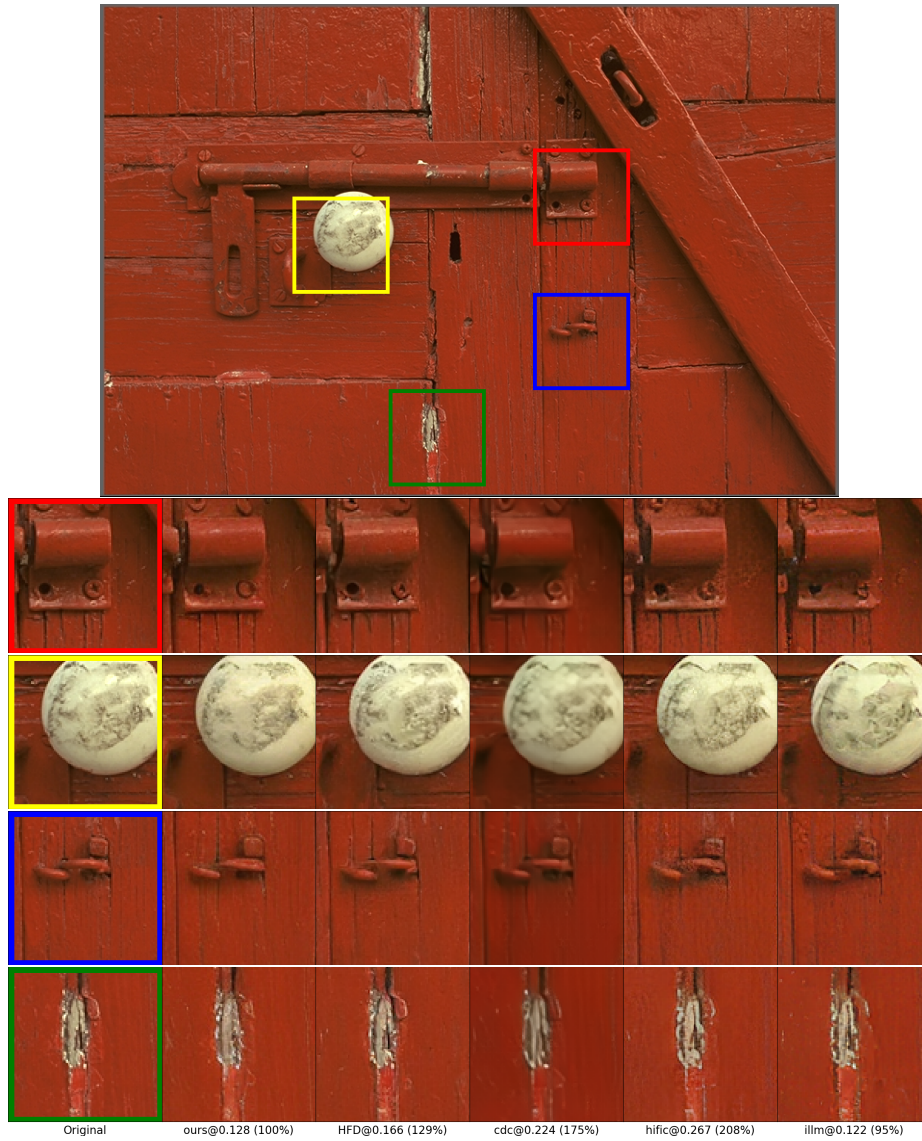


Fig. 13: Additional visual comparisons (kodim02). Images are labeled with Method@bpp (% bpp compared to Ours).



Fig. 14: Additional visual comparisons (kodim05). Images are labeled with Method@bpp (% bpp compared to Ours).

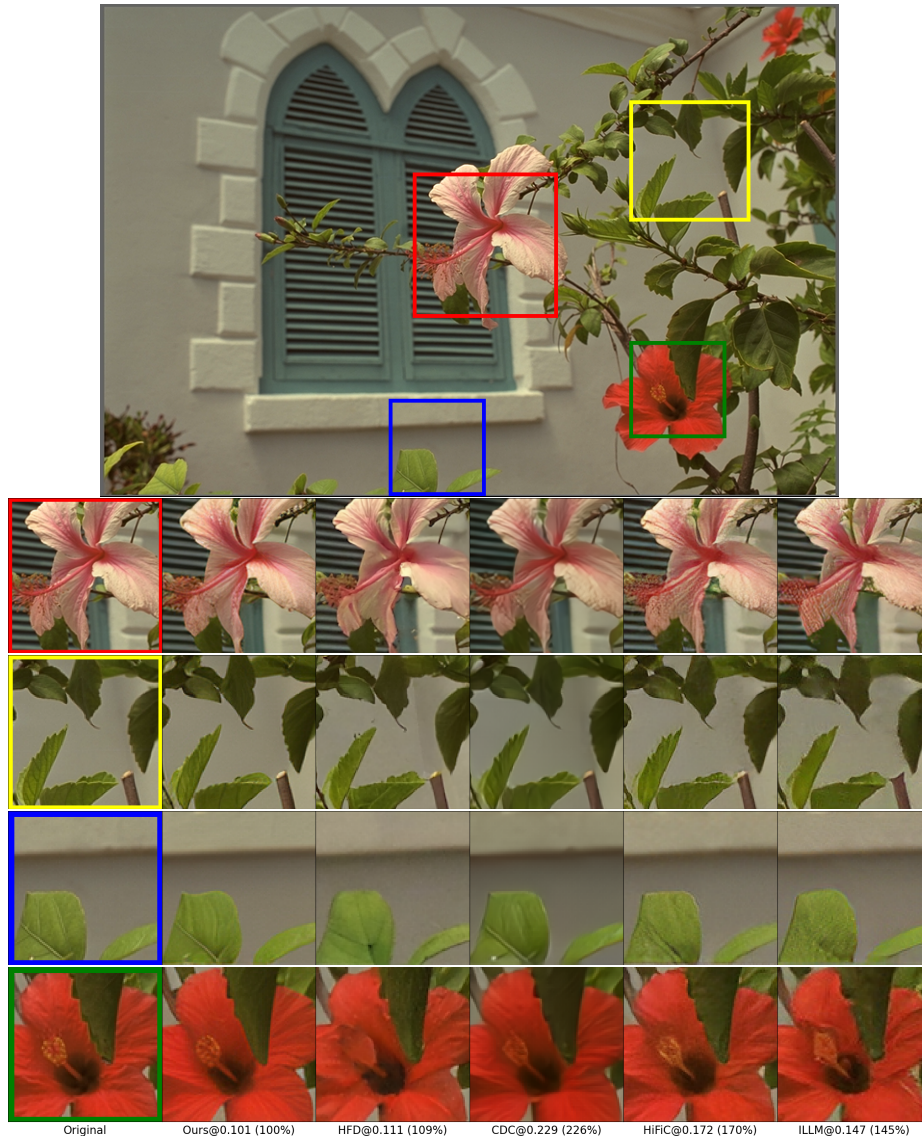


Fig. 15: Additional visual comparisons (kodim07). Images are labeled with Method@bpp (% bpp compared to Ours).

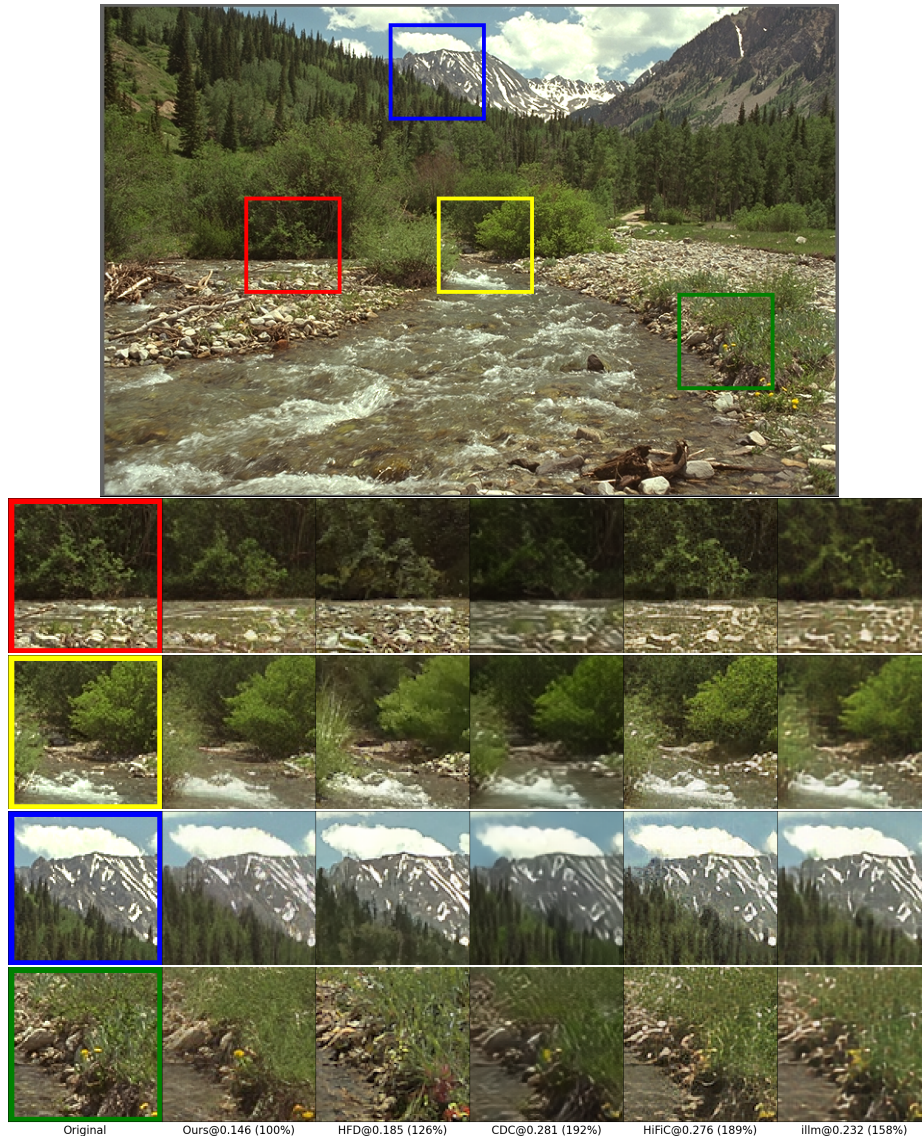


Fig. 16: Additional visual comparisons (kodim13). Images are labeled with Method@bpp (% bpp compared to Ours).

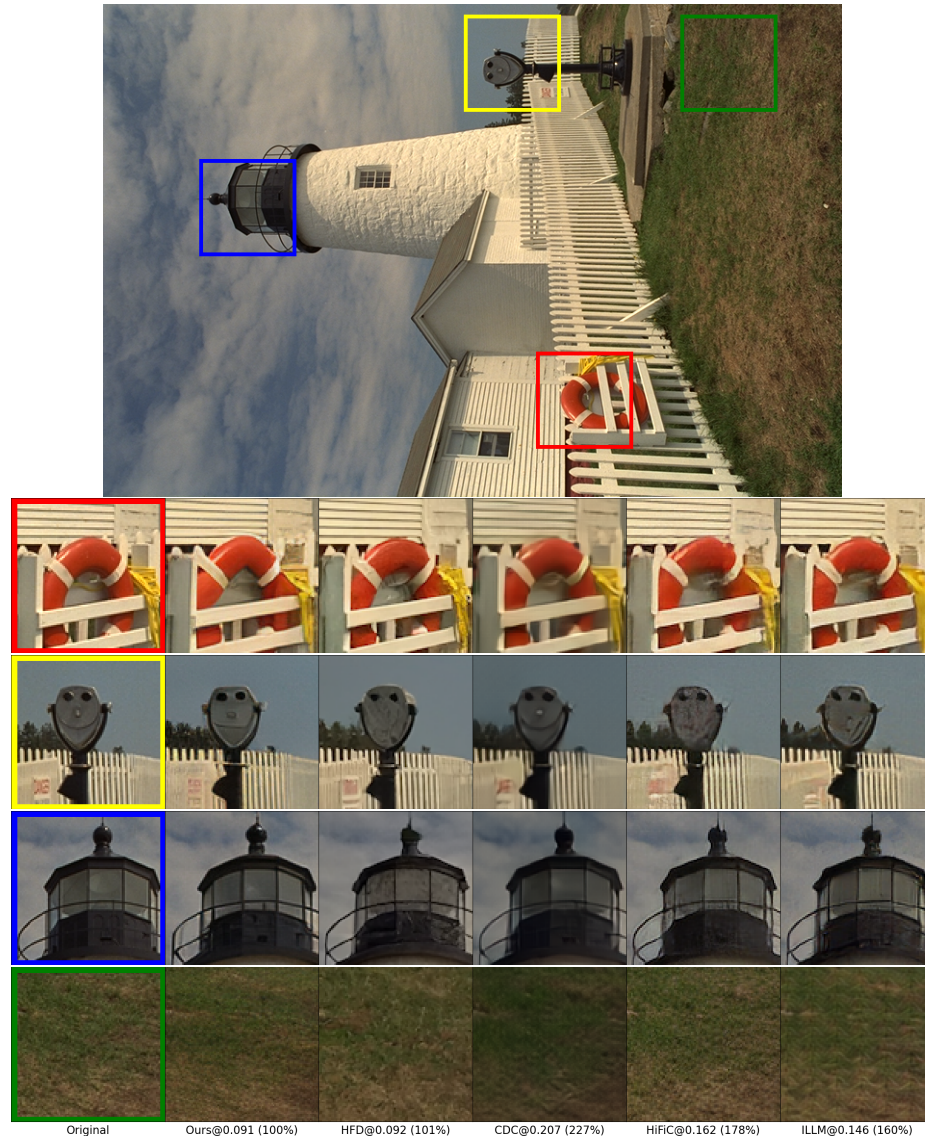


Fig. 17: Additional visual comparisons (kodim19). Images are labeled with Method@bpp (% bpp compared to Ours).



Fig. 18: Additional visual comparisons (kodim24). Images are labeled with Method@bpp (% bpp compared to Ours).

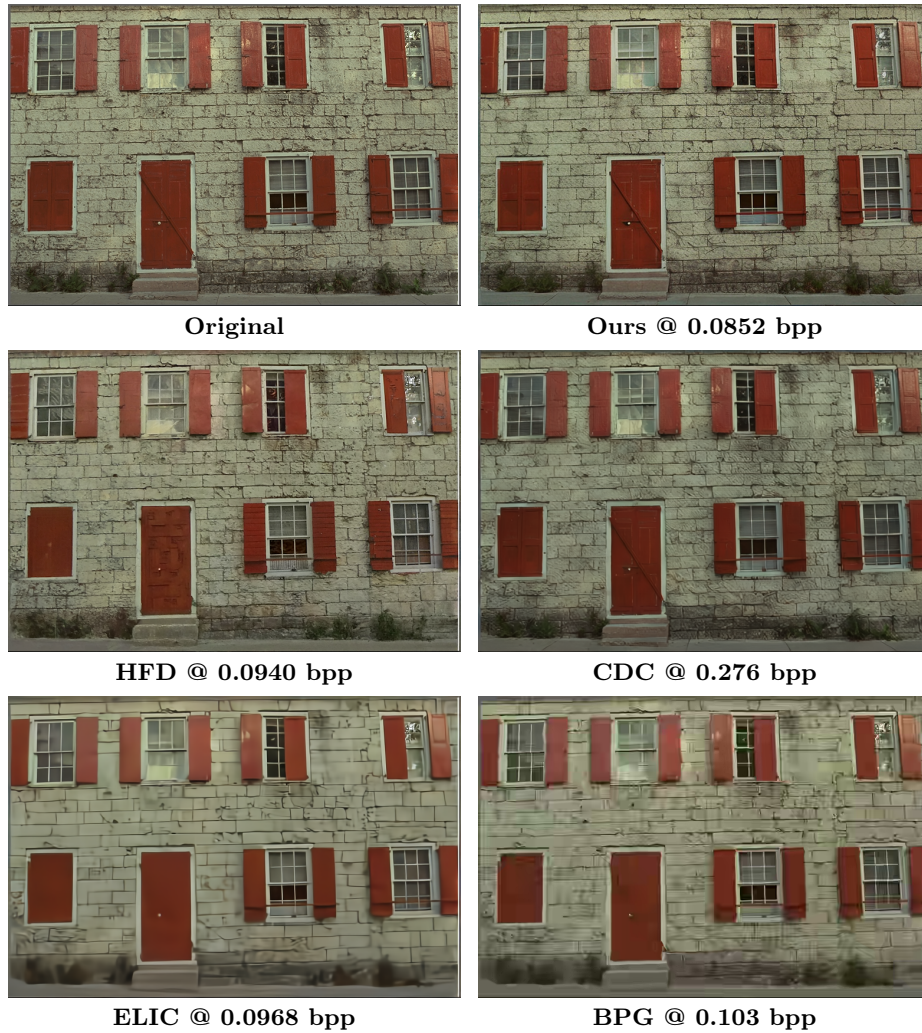


Fig. 19: Qualitative comparison of our method, HFD [25], CDC [46], ELIC [21], and BPG [1] on *kodim01*. Best viewed digitally.

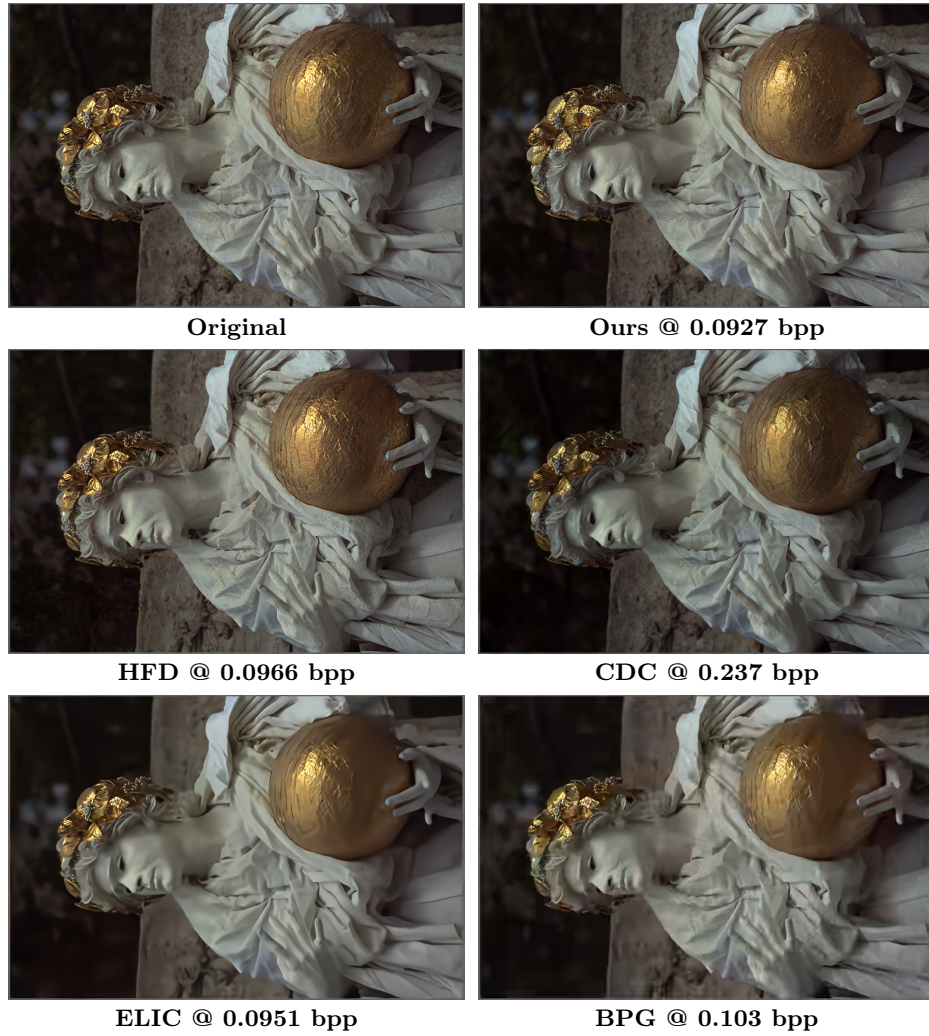


Fig. 20: Qualitative comparison of our method, HFD [25], CDC [46], ELIC [21], and BPG [1] on *kodim17*. Best viewed digitally.

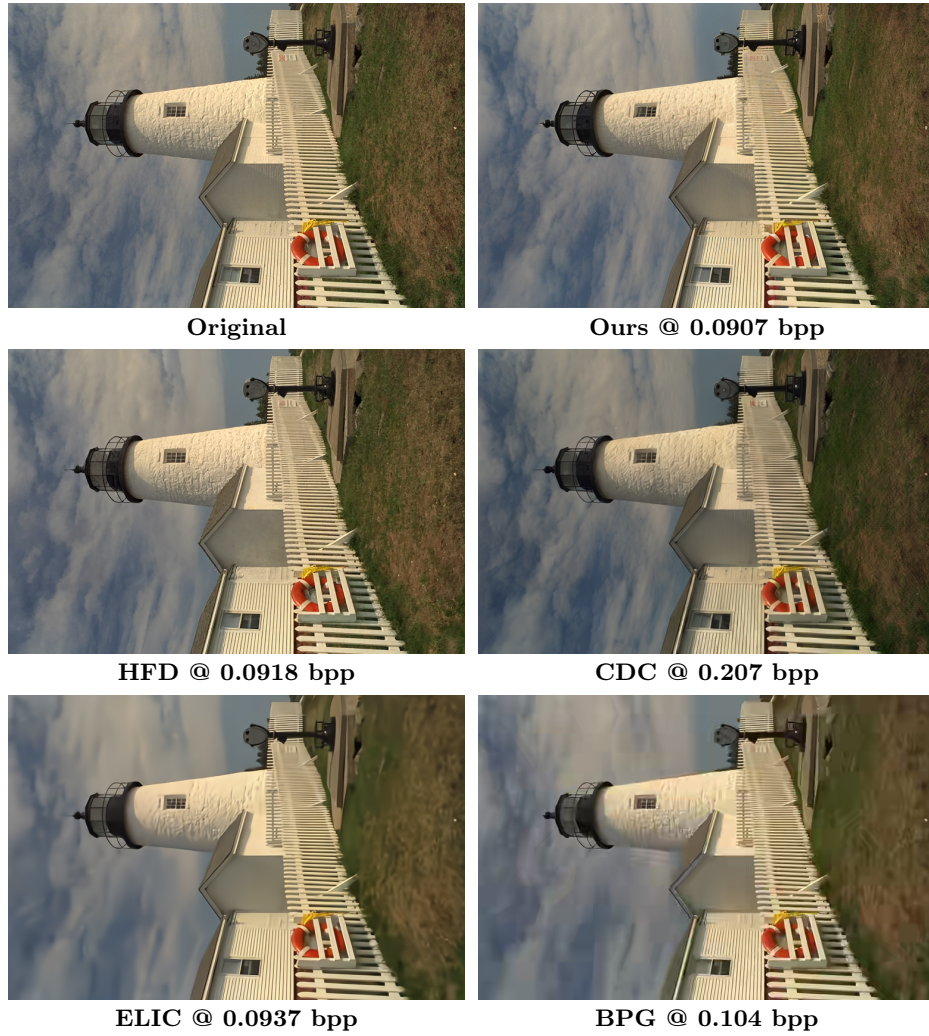


Fig. 21: Qualitative comparison of our method, HFD [25], CDC [46], ELIC [21], and BPG [1] on *kodim19*. Best viewed digitally.



Fig. 22: Qualitative comparison of our method, HFD [25], CDC [46], ELIC [21], and BPG [1] on *kodim23*. Best viewed digitally.



Fig. 23: Qualitative comparison of our method, HFD [25], CDC [46], ELIC [21], and BPG [1] on *kodim24*. Best viewed digitally.

References

1. BPG Image format. <https://bellard.org/bpg/> 1, 2, 18, 30, 31, 32, 33, 34
2. Zlib. <https://www.zlib.net/> 6
3. Challenge on Learned Image Compression (2022) 10
4. Agustsson, E., Minnen, D., Toderici, G., Mentzer, F.: Multi-Realism Image Compression with a Conditional Generator (Mar 2023). <https://doi.org/10.48550/arXiv.2212.13824> 1, 5, 10
5. Agustsson, E., Tschannen, M., Mentzer, F., Timofte, R., Gool, L.V.: Generative Adversarial Networks for Extreme Learned Image Compression. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 221–231 (2019) 1, 2
6. Bachard, T., Bordin, T., Maugey, T.: Coclico: Extremely low bitrate image compression based on clip semantic and tiny color map. In: Picture Coding Symposium 2024 (2024) 2, 4
7. Ballé, J., Laparra, V., Simoncelli, E.P.: End-to-end Optimized Image Compression (Mar 2017). <https://doi.org/10.48550/arXiv.1611.01704> 1, 3, 7, 19
8. Ballé, J., Minnen, D., Singh, S., Hwang, S.J., Johnston, N.: Variational image compression with a scale hyperprior (May 2018). <https://doi.org/10.48550/arXiv.1802.01436> 1, 7, 9, 19
9. Blattmann, A., Rombach, R., Ling, H., Dockhorn, T., Kim, S.W., Fidler, S., Kreis, K.: Align Your Latents: High-Resolution Video Synthesis With Latent Diffusion Models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 22563–22575 (2023) 5
10. Blau, Y., Michaeli, T.: Rethinking Lossy Compression: The Rate-Distortion-Perception Tradeoff. In: Proceedings of the 36th International Conference on Machine Learning. pp. 675–685. PMLR (May 2019) 1, 5, 11
11. Bommasani, R., Hudson, D.A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M.S., Bohg, J., Bosselut, A., Brunskill, E., et al.: On the opportunities and risks of foundation models. arXiv preprint arXiv:2108.07258 (2021) 2
12. Boncelet, C.: Chapter 7 - image noise models. In: Bovik, A. (ed.) The Essential Guide to Image Processing, pp. 143–167. Academic Press, Boston (2009). <https://doi.org/https://doi.org/10.1016/B978-0-12-374457-9.00007-X>, <https://www.sciencedirect.com/science/article/pii/B978012374457900007X> 19
13. Bordin, T., Maugey, T.: Semantic based generative compression of images for extremely low bitrates. In: MMSP 2023 - IEEE 25th International Workshop on MultiMedia Signal Processing. pp. 1–6. IEEE, Poitiers, France (Sep 2023), <https://hal.science/hal-04231421> 2
14. Careil, M., Muckley, M.J., Verbeek, J., Lathuilière, S.: Towards image compression with perfect realism at ultra-low bitrates (Oct 2023). <https://doi.org/10.48550/arXiv.2310.10325> 1, 2, 4, 11
15. Chen, J., Yu, J., Ge, C., Yao, L., Xie, E., Wu, Y., Wang, Z., Kwok, J., Luo, P., Lu, H., Li, Z.: PixArt- α : Fast Training of Diffusion Transformer for Photorealistic Text-to-Image Synthesis (Oct 2023). <https://doi.org/10.48550/arXiv.2310.00426> 2
16. Dhariwal, P., Nichol, A.Q.: Diffusion Models Beat GANs on Image Synthesis. In: Advances in Neural Information Processing Systems (Nov 2021) 2
17. Forsgren, S., Martiros, H.: Riffusion - Stable diffusion for real-time music generation (2022), <https://riffusion.com/about> 2

18. Glickman, M.E.: A Comprehensive Guide to Chess Ratings. *American Chess Journal* **3**(1), 59–102 (1995) [11](#)
19. Goose, N.F., Petersen, J., Wiggers, A., Xu, T., Sautière, G.: Neural Image Compression with a Diffusion-Based Decoder (Jan 2023). <https://doi.org/10.48550/arXiv.2301.05489> [3](#), [4](#)
20. Graikos, A., Malkin, N., Jojic, N., Samaras, D.: Diffusion Models as Plug-and-Play Priors. *Advances in Neural Information Processing Systems* **35**, 14715–14728 (Dec 2022) [2](#)
21. He, D., Yang, Z., Peng, W., Ma, R., Qin, H., Wang, Y.: ELIC: Efficient Learned Image Compression With Unevenly Grouped Space-Channel Contextual Adaptive Coding. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 5718–5727 (2022) [2](#), [18](#), [19](#), [30](#), [31](#), [32](#), [33](#), [34](#)
22. Hendrycks, D., Gimpel, K.: Gaussian error linear units (gelus) (2023) [8](#)
23. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems* **30** (2017) [10](#)
24. Ho, J., Jain, A., Abbeel, P.: Denoising Diffusion Probabilistic Models. In: *Advances in Neural Information Processing Systems*. vol. 33, pp. 6840–6851. Curran Associates, Inc. (2020) [5](#), [9](#)
25. Hoogeboom, E., Agustsson, E., Mentzer, F., Versari, L., Toderici, G., Theis, L.: High-Fidelity Image Compression with Score-based Generative Models (May 2023). <https://doi.org/10.48550/arXiv.2305.18231> [1](#), [2](#), [3](#), [4](#), [5](#), [10](#), [11](#), [18](#), [30](#), [31](#), [32](#), [33](#), [34](#)
26. Hoogeboom, E., Heek, J., Salimans, T.: Simple diffusion: End-to-end diffusion for high resolution images. In: *Proceedings of the 40th International Conference on Machine Learning. ICML'23*, vol. 202, pp. 13213–13232. JMLR.org, Honolulu, Hawaii, USA (Jul 2023) [14](#)
27. Jabri, A., Fleet, D.J., Chen, T.: Scalable adaptive computation for iterative generation. In: *Proceedings of the 40th International Conference on Machine Learning. ICML'23*, vol. 202, pp. 14569–14589. JMLR.org, Honolulu, Hawaii, USA (Jul 2023) [14](#)
28. Ke, B., Obukhov, A., Huang, S., Metzger, N., Dautt, R.C., Schindler, K.: Repurposing Diffusion-Based Image Generators for Monocular Depth Estimation (Dec 2023). <https://doi.org/10.48550/arXiv.2312.02145> [2](#)
29. Kingma, D.P., Welling, M.: Auto-Encoding Variational Bayes. In: *2nd International Conference on Learning Representations, ICLR (2014)* [5](#)
30. Kodak: PhotoCD PCD0992 (1993) [10](#)
31. Lei, E., Uslu, Y.B., Hassani, H., Bidokhti, S.S.: Text + Sketch: Image Compression at Ultra Low Rates (Jul 2023). <https://doi.org/10.48550/arXiv.2307.01944> [2](#), [4](#)
32. Meng, C., He, Y., Song, Y., Song, J., Wu, J., Zhu, J.Y., Ermon, S.: SDEdit: Guided Image Synthesis and Editing with Stochastic Differential Equations. In: *International Conference on Learning Representations (2022)* [3](#)
33. Mentzer, F., Toderici, G., Tschannen, M., Agustsson, E.: High-Fidelity Generative Image Compression (Oct 2020). <https://doi.org/10.48550/arXiv.2006.09965> [1](#), [2](#), [10](#), [11](#)
34. Minnen, D., Ballé, J., Toderici, G.: Joint Autoregressive and Hierarchical Priors for Learned Image Compression (Sep 2018). <https://doi.org/10.48550/arXiv.1809.02736> [1](#), [7](#), [9](#), [19](#)

35. Muckley, M.J., El-Nouby, A., Ullrich, K., Jegou, H., Verbeek, J.: Improving Statistical Fidelity for Neural Image Compression with Implicit Local Likelihood Models. In: Proceedings of the 40th International Conference on Machine Learning. pp. 25426–25443. PMLR (Jul 2023) [2](#), [11](#)
36. Pasco, R.C.: Source coding algorithms for fast data compression (1976) [9](#)
37. Qian, Y., Lin, M., Sun, X., Tan, Z., Jin, R.: Entroformer: A Transformer-based Entropy Model for Learned Image Compression (Mar 2022). <https://doi.org/10.48550/arXiv.2202.05492> [9](#), [18](#), [19](#)
38. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-Resolution Image Synthesis With Latent Diffusion Models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10684–10695 (2022) [2](#), [5](#), [6](#), [7](#)
39. Salimans, T., Ho, J.: Progressive Distillation for Fast Sampling of Diffusion Models. In: International Conference on Learning Representations (Jun 2022). <https://doi.org/10.48550/arXiv.2202.00512> [18](#)
40. Shannon, C.E.: A Mathematical Theory of Communication. The Bell System Technical Journal **27**, 379–423 (1948) [4](#)
41. Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., Ganguli, S.: Deep Unsupervised Learning using Nonequilibrium Thermodynamics. In: Proceedings of the 32nd International Conference on Machine Learning. pp. 2256–2265. PMLR (Jun 2015) [5](#)
42. Song, J., Meng, C., Ermon, S.: Denoising Diffusion Implicit Models. In: International Conference on Learning Representations (Jan 2021) [9](#)
43. Stein, G., Cresswell, J.C., Hosseinzadeh, R., Sui, Y., Ross, B.L., Villedcroze, V., Liu, Z., Caterini, A.L., Taylor, J.E.T., Loaiza-Ganem, G.: Exposing flaws of generative model evaluation metrics and their unfair treatment of diffusion models (Oct 2023). <https://doi.org/10.48550/arXiv.2306.04675> [11](#)
44. Wallace, G.: The JPEG still picture compression standard. IEEE Transactions on Consumer Electronics **38**(1), xviii–xxxiv (1992). <https://doi.org/10.1109/30.125072> [1](#)
45. Xue, T., Chen, B., Wu, J., Wei, D., Freeman, W.T.: Video Enhancement with Task-Oriented Flow. International Journal of Computer Vision **127**(8), 1106–1125 (Aug 2019). <https://doi.org/10.1007/s11263-018-01144-2> [9](#)
46. Yang, R., Mandt, S.: Lossy Image Compression with Conditional Diffusion Models. Advances in Neural Information Processing Systems **36**, 64971–64995 (Dec 2023) [2](#), [3](#), [11](#), [18](#), [30](#), [31](#), [32](#), [33](#), [34](#)
47. Zhang, L., Rao, A., Agrawala, M.: Adding Conditional Control to Text-to-Image Diffusion Models. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 3836–3847 (2023) [2](#)