Bridging the Gap between Diffusion Models and Universal Quantization for Image Compression

Lucas Relic[1,2] **Roberto Azevedo**[2] **Yang Zhang**[2] **Markus Gross**[1,2] **Christopher Schroers**[2]
[1]ETH Zürich [2]DisneyResearch|Studios
{lucas.relic, grossm}@inf.ethz.ch
{roberto.azevedo, yang.zhang, christopher.schroers}@disneyresearch.com

Abstract

By leveraging the similarities between quantization error and additive noise, diffusion-based image compression codecs can be built by using a diffusion model to "denoise" the artifacts introduced by quantization. However, we identify three gaps in this approach which result in the quantized data falling out of distribution of the diffusion model: a gap in noise level, noise type, and a gap caused by discretization. To address these issues, we propose a novel quantization-based forward diffusion process that is theoretically founded and bridges all three aforementioned gaps. This is achieved through universal quantization with a carefully tailored quantization schedule, as well as diffusion model trained for uniform noise. Compared to previous work, our proposed architecture produces consistently realistic and detailed results, even at extremely low bitrates, while maintaining strong faithfulness to the original images.

1 Introduction

In today's increasingly data-hungry society, the field of neural image compression (NIC) has seen remarkable growth in both the demand and effectiveness of codecs. A critical step in lossy NIC is quantization, where continuous latent representations must be discretized to encode the data into a bitstream for transmission. This negatively affects reconstruction quality, as quantization introduces error into the signal, and also produces a discrete variable which does not lie in the continuous distribution the neural decoder was trained for. As a result, most NIC codecs address this issue by building decoders that are robust to such errors.

Methods based on the recently proposed diffusion models [1; 2] follow a similar paradigm. Here, the powerful generative capability of diffusion models is harnessed to produce highly detailed, realistic reconstructions, especially at extremely low bitrates. This is achieved by conditioning the diffusion model on information from the source image, and generating a new image which attempts to match the source as closely as possible. The conditioning often takes the form of some spatial information, such as learned embedding [3; 4], an image compressed via another codec [5; 6], or an edge or color map [7; 8]. It can also additionally be an unstructured content variable, for example text from an image captioning model [4] or a CLIP embedding [7; 8]. Regardless of modality, constructing a robust diffusion-based decoder and conditionally sampling an image requires a long decoding time, due to the iterative diffusion process, and often requires training a diffusion model from scratch to accept the desired conditioning modality.

However, diffusion models also allow the problem to be approached from a different perspective. Quantization error is often modeled as noise [9; 10], and given that diffusion models are denoising models, one can directly apply them to the data to remove quantization artifacts. Relic et al. [11] first introduced such a pipeline and proposed a learned module that selectively quantizes data (i.e., introduces noise) which is then denoised by the diffusion model at the receiver. However, we

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

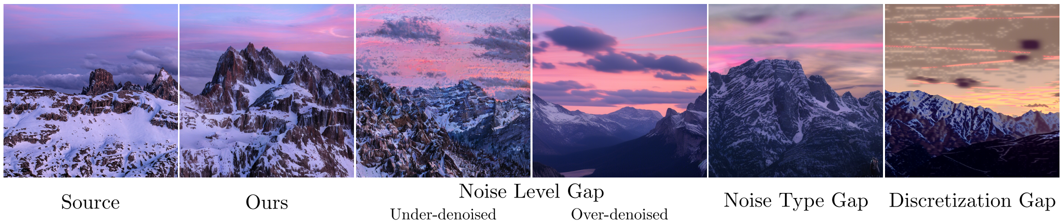

Figure 1: Visualization of the 3 gaps we address in this work. Failure to match the noise level (middle columns) results in either too noisy or too smooth images. Inconsistent noise types (middle-right) introduces generative artifacts and color shift. Applying diffusion to discrete data (far right) causes flat textures as well as color shift. Addressing all three gaps (middle-left) results in the most realistic reconstruction that best matches the source image (far left).

identify three gaps in their approach: the *noise level gap*, the *noise type gap*, and the *discretization gap*. The *noise level gap* represents a difference in the amount of noise introduced via quantization versus the amount of noise expected at the given diffusion step. Diffusion models are sensitive to such mismatches (Fig. 1, middle) and produce either noisy or over-smoothed reconstructions in the presence of this gap. The *noise type gap* is a gap in the class of distribution from which the noise is drawn from. In this scenario, quantization error is well approximated by a uniform distribution [9], while diffusion models are trained on Gaussian noise, resulting in the noise type gap (Fig. 1, middle-right). The *discretization gap* signifies a mismatch when passing quantized data to a neural network which was trained on continuous data. Small variations in the data are eliminated during discretization, and many values can be quantized to the same bin, which results in flat texures or color shifts in the image reconstructions (Fig. 1, far right).

We propose a pipeline that addresses and bridges all the aforementioned gaps. To achieve this, we introduce a new quantization-based diffusion forward process that theoretically aligns with the original forward process, placing the quantized data perfectly along the diffusion trajectory. Our proposed forward process utilizes universal quantization to close the discretization gap, and we introduce a quantization schedule that dictates the signal-to-noise ratio of the quantized data, which can be aligned with a diffusion model to close the noise level gap. Finally, we resolve the noise type gap by using a diffusion model trained with uniform noise, therefore matching the distribution of quantization error. We additionally show that such a uniform noise diffusion model can be obtained in a computationally efficient manner by fine-tuning existing foundation Gaussian diffusion models. Using this forward process, we build a diffusion-based compression codec and show that it produces more realistic and detailed reconstructions than other methods and can operate at a wider range of target bitrates.

2 Method

The basis of our approach is a theoretically founded connection between quantization error and the diffusion process. Such a link allows a diffusion model to be used as a compression codec by "denoising" quantization artifacts at the reciever [11]. We achieve this by proposing a new quantization-based forward diffusion process and implementing it within a latent diffusion model.

2.1 Universal quantization diffusion compression

Central to our method is a diffusion model [1; 12], which defines a process that models the transition between random noise and structured data. When the forward (data to noise) and reverse (noise to data) processes are divided into small steps, the transition between each step is the addition or removal of a Gaussian noise sample. The full diffusion process is thus a traversal between a series of timesteps $t \in [N, 0]$. While this process is iterative, one can also express the partially noisy diffusion variable $\mathbf{y}_t$ at any given t in terms of the original data $\mathbf{y}_0$ and a noise sample ϵ:

$$\mathbf{y}_t = \sqrt{\bar{\alpha}_t}\mathbf{y}_0 + \epsilon, \quad \epsilon \sim \mathcal{N}(0, (1 - \bar{\alpha}_t)\mathbf{I}), \tag{1}$$

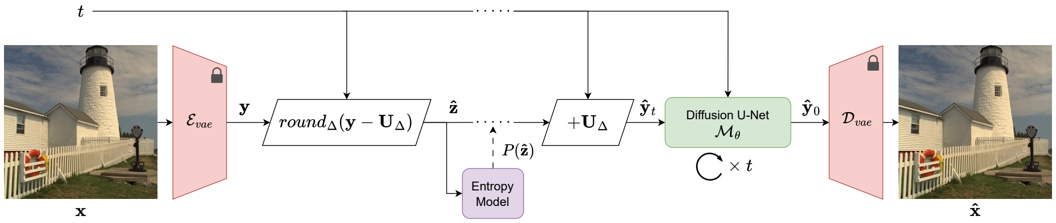

Figure 2: Architecture of our proposed method. An input image is first encoded to the latent space of a diffusion model and passed through the pre-quantization stage of our proposed forward process. The discretized data is transmitted across the channel, subject to the post-quantization stage of Eq. 4, denoised by the diffusion model, and decoded back to image space. The user-input timestep parameter dictates the quantization bin size and variance of noise used in the forward process, as well as the number of denoising steps performed by the diffusion model.

where $\bar{\alpha}_t$ (known as the "variance schedule") defines the ratio of signal and noise at every t and increases as $t \to 0$. In reality, the reverse process is intractable and thus parameterized by the diffusion model, which learns to iteratively denoise $\mathbf{y}_t$ by stepping through $t = \{N, ..., 1, 0\}$.

We propose to replace the forward diffusion process with universal quantization. Universal quantization [13; 14] is defined as hard quantization dithered by a uniform random variable. This has the unique property of being equal in distribution to simply adding another sample from an identical random variable to the original unquantized data [15]:

$$\hat{\mathbf{y}} = \lfloor \mathbf{y} - \mathbf{u} \rceil_\Delta + \mathbf{u} \overset{d}{=} \mathbf{y} + \mathbf{u}', \quad \mathbf{u}, \mathbf{u}' \sim \mathcal{U}[-\Delta/2, \Delta/2], \tag{2}$$

where $\lfloor \cdot \rceil_\Delta$ denotes quantization to a bin of width Δ. We define a new diffusion forward process by combining Eqs. 1 and 2, which is separated into pre- and post-quantization stages:

$$\hat{\mathbf{y}}_t = \lfloor \sqrt{\bar{\alpha}_t}\mathbf{y} - \mathbf{u} \rceil_\Delta + \mathbf{u}, \quad \mathbf{u} \sim \mathcal{U}[-\Delta/2, \Delta/2] \tag{3}$$

$$\hat{\mathbf{z}} = \lfloor \sqrt{\bar{\alpha}_t}\mathbf{y} - \mathbf{u} \rceil_\Delta, \quad \hat{\mathbf{y}}_t = \hat{\mathbf{z}} + \mathbf{u}. \tag{4}$$

Critically, for Eq. 3 to be a suitable substitute for Eq. 1, we must ensure $\hat{\mathbf{y}}_t$ follows the same variance schedule as $\mathbf{y}_t$. We achieve this by proposing a *quantization schedule* which dictates the quantization bin width and uniform variable support, thus controlling the SNR of $\hat{\mathbf{y}}_t$. Specifically, setting $\Delta = \sqrt{12(1 - \bar{\alpha}_t)}$ in Eq. 3 aligns our quantization schedule with the diffusion variance schedule.[1]

Our proposed quantization-based forward process simultaneously eliminates both the noise level and discretization gap, via the quantization schedule and the use of universal quantization, respectively. An added benefit of our quantization schedule is that t also becomes a rate-distortion tradeoff parameter, as the quantization bin width has a direct impact on the final size of the compressed bitstream. Additionally, because diffusion models can denoise data at any arbitrary timestep, our method supports compression to multiple bitrates with a single model by accepting t as user input at inference time.

2.2 Uniform noise diffusion model

To address the noise type gap, we use a diffusion model trained for uniform noise, which matches the distribution of error introduced during quantization. Several works [16; 17; 18] investigate diffusion models under varying noise (and even data [19]) distributions, with Heitz et al. [18] proposing a diffusion formulation which is valid for any distribution of finite variance, for which the uniform distribution qualifies. However, training such a diffusion model (either Gaussian or non-Gaussian) from scratch requires significant resources [20] and can be computationally prohibitive. We find that existing foundation Gaussian diffusion models can be applied to uniform noise by finetuning the

[1] We provide a derivation in Appendix B.

model on the desired distribution. A critical operation in this process, as in Sec. 2.1, is ensuring the signal-to-noise ratio of the noisy diffusion variable under the new distribution follows the variance schedule of the original diffusion model. In our scenario of adapting a Gaussian diffusion model to uniform noise, this is done by drawing $\epsilon \sim \mathcal{U}\left(-\sqrt{3(1-\bar{\alpha}_t)}, \sqrt{3(1-\bar{\alpha}_t)}\right)$ in Eq. 1.

2.3 Implementation

We implement our quantization strategy with a latent diffusion model because quantization in this latent space is uniformly distributed, which is not necessarily the case in the image domain. Fig. 2 shows an overview of our pipeline. An input image is first encoded to latent space via the VAE encoder. We then apply only the pre-quantization stage of our proposed forward process (Eq. 4), which produces a discrete latent that is coded to a bitstream using a learned entropy model. At the receiver, the post-quantization stage is performed and the latent passed to the diffusion model, which reconstructs information lost during quantization. Finally, the reconstructed latent is decoded back to image space.

We implement our pipeline using Stable Diffusion v2.1 [21] and follow the same training procedure for finetuning, except for using to Uniform noise of proper variance instead of Gaussian noise. Our model is trained for 100k steps on a subset of the LAION Improved Aesthetics 6.5+ dataset[2] with batch size 8 and learning rate 1e-5. We additionally employ input perturbation [22] as this has been shown to improve sampling quality. The VAE encoder and decoder are not finetuned.

Entropy coding is performed with a mean-scale hyperprior entropy model [23] and arithmetic coding (implemented via CompressAI [24]). Because the latent transform and quantization process are fixed, we optimize the entropy model only on the rate objective. Both the VAE and diffusion model are frozen when training the entropy model.

3 Experiments

We validate the performance of our method via qualitative comparison of compressed images. The codec proposed by **Relic et al.** [11] is used as a baseline as it is most similar to our method, and we evaluate on the **Kodak** dataset. Our method consistently produces more realistic and detailed reconstructions, shown in Fig. 3. This can be seen in the red wood or cloud (top row), where Relic et al. contains flat textures (top right) and unnatural color shifts (top right and top left) that do not appear in our reconstructions. Additionally, our reconstructions are always realistic, compared to the baseline which can produce over-smoothed images (bottom right).

We report additional qualitative comparisons to other compression baselines, as well as quantitative metrics, in Appendix A.

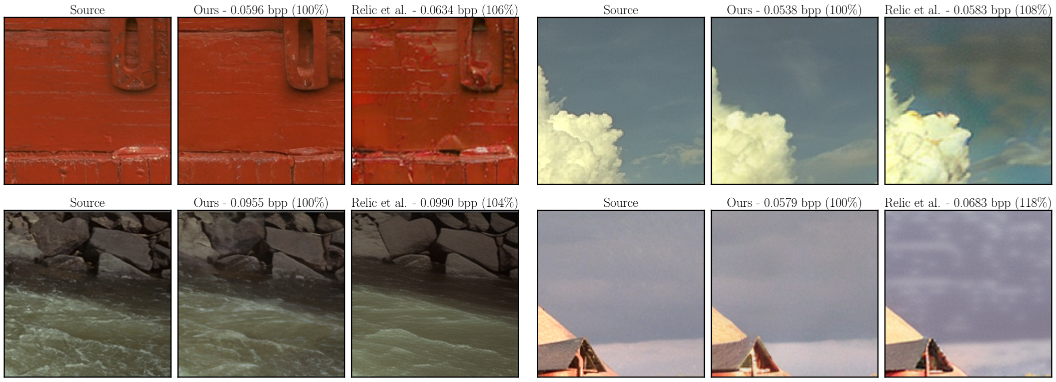
Source
Ours - 0.0596 bpp (100%)
Relic et al. - 0.0634 bpp (106%)
Source
Ours - 0.0538 bpp (100%)
Relic et al. - 0.0583 bpp (108%)
Source
Ours - 0.0955 bpp (100%)
Relic et al. - 0.0990 bpp (104%)
Source
Ours - 0.0579 bpp (100%)
Relic et al. - 0.0683 bpp (118%)

Figure 3: Qualitative comparison of our method to Relic et al. [11]. Best viewed zoomed in.

[2] https://huggingface.co/datasets/bhargavsdesai/laion_improved_aesthetics_6.5plus_with_images

References

[1] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli, “Deep Unsupervised Learning using Nonequilibrium Thermodynamics,” in *Proceedings of the 32nd International Conference on Machine Learning*, pp. 2256–2265, PMLR, June 2015.

[2] J. Ho, A. Jain, and P. Abbeel, “Denoising Diffusion Probabilistic Models,” in *Advances in Neural Information Processing Systems*, vol. 33, pp. 6840–6851, Curran Associates, Inc., 2020.

[3] R. Yang and S. Mandt, “Lossy Image Compression with Conditional Diffusion Models,” *Advances in Neural Information Processing Systems*, vol. 36, pp. 64971–64995, Dec. 2023.

[4] M. Careil, M. J. Muckley, J. Verbeek, and S. Lathuilière, “Towards image compression with perfect realism at ultra-low bitrates,” Oct. 2023.

[5] E. Hoogeboom, E. Agustsson, F. Mentzer, L. Versari, G. Toderici, and L. Theis, “High-Fidelity Image Compression with Score-based Generative Models,” May 2023.

[6] N. F. Goose, J. Petersen, A. Wiggers, T. Xu, and G. Sautière, “Neural Image Compression with a Diffusion-Based Decoder,” Jan. 2023.

[7] E. Lei, Y. B. Uslu, H. Hassani, and S. S. Bidokhti, “Text + Sketch: Image Compression at Ultra Low Rates,” July 2023.

[8] T. Bachard, T. Bordin, and T. Maugey, “CoCliCo: Extremely low bitrate image compression based on CLIP semantic and tiny color map,” in *PCS 2024 - Picture Coding Symposium*, p. 1, June 2024.

[9] R. Gray and D. Neuhoff, “Quantization,” *IEEE Transactions on Information Theory*, vol. 44, pp. 2325–2383, Oct. 1998.

[10] J. Ballé, V. Laparra, and E. P. Simoncelli, “End-to-end Optimized Image Compression,” in *International Conference on Learning Representations*, Feb. 2017.

[11] L. Relic, R. Azevedo, M. Gross, and C. Schroers, “Lossy Image Compression with Foundation Diffusion Models,” in *Proceedings of the IEEE/CVF European Conference on Computer Vision (ECCV)*, September 2024.

[12] J. Song, C. Meng, and S. Ermon, “Denoising Diffusion Implicit Models,” in *International Conference on Learning Representations*, Jan. 2021.

[13] J. Ziv, “On universal quantization,” *IEEE Transactions on Information Theory*, vol. 31, pp. 344–347, May 1985.

[14] R. Zamir and M. Feder, “On universal quantization by randomized uniform/lattice quantizers,” *IEEE Transactions on Information Theory*, vol. 38, pp. 428–436, Mar. 1992.

[15] E. Agustsson and L. Theis, “Universally Quantized Neural Compression,” in *Advances in Neural Information Processing Systems*, vol. 33, pp. 12367–12376, Curran Associates, Inc., 2020.

[16] J. Deasy, N. Simidjievski, and P. Liò, “Heavy-tailed denoising score matching,” Apr. 2022.

[17] E. Nachmani, R. S. Roman, and L. Wolf, “Denoising Diffusion Gamma Models,” Oct. 2021.

[18] E. Heitz, L. Belcour, and T. Chambon, “Iterative α-(de)Blending: A Minimalist Deterministic Diffusion Model,” May 2023.

[19] A. Bansal, E. Borgnia, H.-M. Chu, J. S. Li, H. Kazemi, F. Huang, M. Goldblum, J. Geiping, and T. Goldstein, “Cold Diffusion: Inverting Arbitrary Image Transforms Without Noise,” Aug. 2022.

[20] J. Chen, J. Yu, C. Ge, L. Yao, E. Xie, Y. Wu, Z. Wang, J. Kwok, P. Luo, H. Lu, and Z. Li, “PixArt-α: Fast Training of Diffusion Transformer for Photorealistic Text-to-Image Synthesis,” Oct. 2023.

[21] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, “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.

[22] M. Ning, E. Sangineto, A. Porrello, S. Calderara, and R. Cucchiara, “Input perturbation reduces exposure bias in diffusion models,” in *Proceedings of the 40th International Conference on Machine Learning* (A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, and J. Scarlett, eds.), vol. 202 of *Proceedings of Machine Learning Research*, pp. 26245–26265, PMLR, 23–29 Jul 2023.

[23] D. Minnen, J. Ballé, and G. D. Toderici, “Joint Autoregressive and Hierarchical Priors for Learned Image Compression,” in *Advances in Neural Information Processing Systems*, vol. 31, Curran Associates, Inc., 2018.

[24] J. Bégaint, F. Racapé, S. Feltman, and A. Pushparaja, “CompressAI: A PyTorch library and evaluation platform for end-to-end compression research,” Nov. 2020.

[25] M. J. Muckley, A. El-Nouby, K. Ullrich, H. Jegou, and J. Verbeek, “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, July 2023.

[26] F. Mentzer, G. D. Toderici, M. Tschannen, and E. Agustsson, “High-fidelity generative image compression,” in *Advances in Neural Information Processing Systems* (H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, eds.), vol. 33, pp. 11913–11924, Curran Associates, Inc., 2020.

[27] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, “Gans trained by a two time-scale update rule converge to a local nash equilibrium,” in *Advances in Neural Information Processing Systems* (I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, eds.), vol. 30, Curran Associates, Inc., 2017.

[28] Y. Blau and T. Michaeli, “Rethinking Lossy Compression: The Rate-Distortion-Perception Tradeoff,” in *Proceedings of the 36th International Conference on Machine Learning*, pp. 675–685, PMLR, May 2019.

Supplementary Material

A Additional Results

In addition to Relic et al. [11], we take the method proposed by Hoogeboom et al. (**HFD**) [5] and Yang and Mandt (**CDC**) [3] to compare against diffusion-based methods. We only include quantitative comparisons against CDC as it operates in a significantly higher bitrate range, and thus a fair qualitative comparison is not feasible. We also compare to **ILLM** [25] and **HiFiC** [26], both GAN-based approaches, to provide a baseline to other generative image compression codecs.

Further qualitative comparions are shown in Fig. 5. Our method consistently produces more realistic reconstructions compared to the baselines. At low bitrates, where other codecs lack detail (rows 1 and 3) or introduce generative artifacts (row 2), our reconstructions are detailed, more closely match the source image, and contain plausible textures. Our proposed method additionally performs well at higher bitrates, where the baselines fail to match the level of detail of our reconstructions (row 4).

For quantitative comparison, we evaluate all baselines using four metrics: **FID** [27], to measure the realism of reconstructed images; **LPIPS**, as a perceptually-oriented pixelwise distortion metric; and **PSNR** and **MS-SSIM** which are metrics traditionally used in image compression evaluation. It is important to note that at low bitrates, PSNR and MS-SSIM do not accurately reflect the quality of reconstructed images [28; 4]. In fact, achieving high performance in pixelwise distortion necessarily decreases visual quality [28]. As one of our goals is to produce realistic and perceptually pleasing images, we do not focus on performance measured by these metrics. Experiments are performed on the Kodak dataset. However, as computing FID metrics requires a large number of images and is thus not possible on the Kodak dataset, we use **MS-COCO 30k** [5] for this comparison.

Quantitative performance is shown in Fig. 4. In rate-realism, our method outperforms all baselines except for Relic et al., for which it remains competitive. At higher bitrates, where Relic et al. is unable to operate, our method achieves the best performance. When measured in LPIPS, our proposed method outperforms all other diffusion-based codecs at nearly all bitrates. Performance of our method suffers when evaluating on PSNR and MS-SSIM due to reasons mentioned above.

Notably, our method can operate at a significantly wider range of bitrates than the baselines. At the lower end, our method can compress images to nearly an order of magnitude fewer bits with reasonable faithfulness to the source image. We additionally can compress to significantly higher bitrates than Relic et al., highlighting the versatility and applicability of our proposal.

To provide examples of our method's ability to consistently produce realistic and plausible images, we show examples of images compressed over a range of bitrates in Fig. 6. Here it can be seen that at high bitrates, the reconstructions are accurate at a pixel level to the source image. As bitrate decreases, the images maintain high realism and semantic alignment and vary in low-level details (*e.g.*, bush, rock, or brick textures), rather than blurring artifacts traditionally seen in neural image compression.

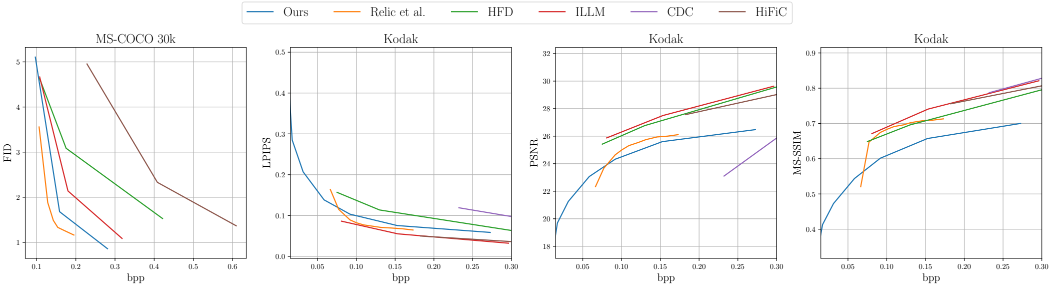

Figure 4: Rate-distortion performance of our method compared to Relic et al. [11], HFD [5], ILLM [25], CDC [3], and HiFiC [26].

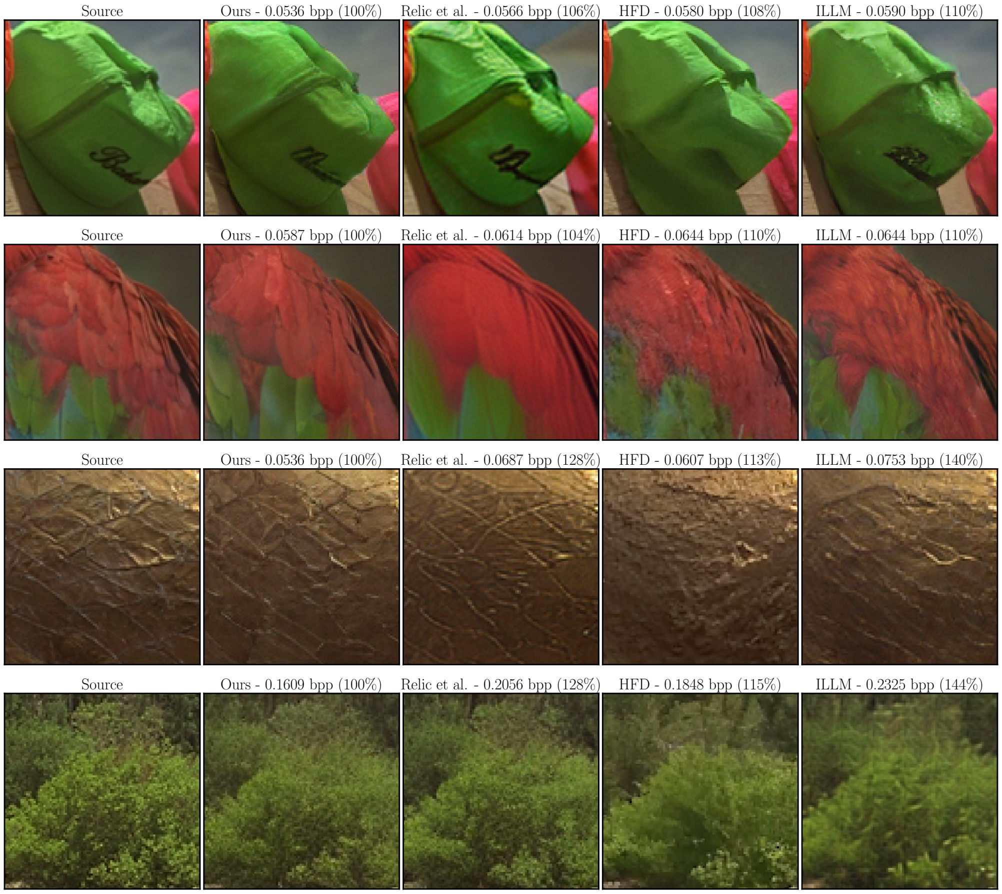

Figure 5: Qualitative comparison of our method to Relic et al. [11], HFD [5], and ILLM [25] on the Kodak dataset. Bitrates are also expressed as a percentage of our method. Best viewed digitally.

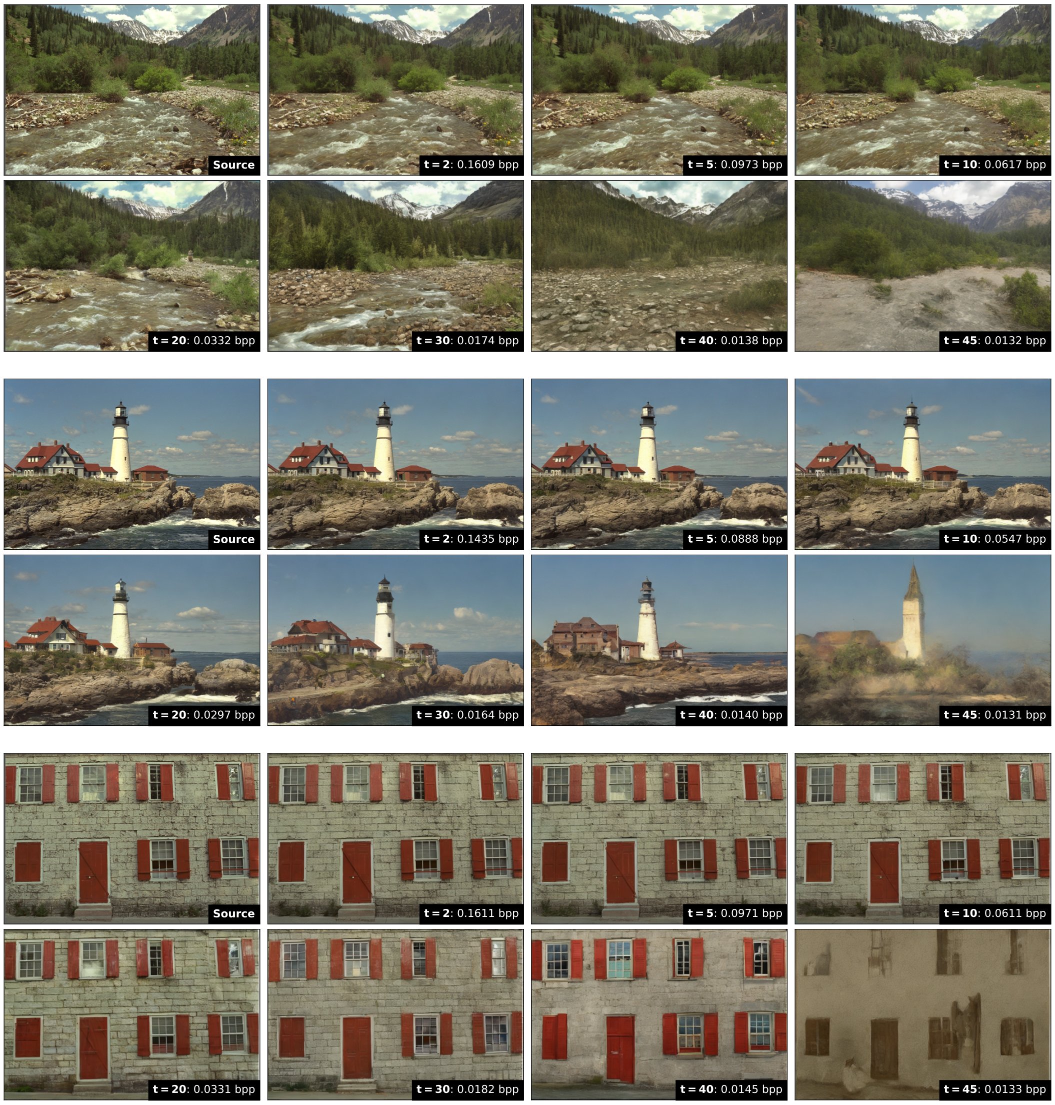

Figure 6: Visual example of image reconstructions produced over a range of bitrates. At high bitrate, our method produces reconstructions with high fidelity to the source image. As bitrate decreases, the images maintain a high realism and semantic alignment, tending towards differences in the spatial alignment of content. The maximum timestep value of the diffusion sampler in this experiment is $t = 50$. Best viewed digitally.

B Matching Signal-to-Noise Ratio via Quantization Schedule

In this section we provide a derivation of our quantization schedule such that the signal-to-noise ratio of the noisy diffusion variable produced by our forward process matches that of the original diffusion process.

The standard Gaussian diffusion process is defined as:

$$\mathbf{y}_t = \sqrt{\bar{\alpha}_t}\mathbf{y}_0 + \epsilon, \quad \epsilon \sim \mathcal{N}(0, (1-\bar{\alpha}_t)\mathbf{I}). \tag{5}$$

In this context, the signal to noise ratio of the noisy variable $\mathbf{y}_t$ is:

$$\text{SNR}_{\mathbf{y}_t} := \frac{\mathbb{E}[S^2]}{\text{Var}[N]} = \frac{\mathbb{E}[(\sqrt{\bar{\alpha}_t}\mathbf{y}_0)^2]}{1-\bar{\alpha}_t}. \tag{6}$$

Using our proposed forward noising process in Eq. 3 and deriving the SNR of $\hat{\mathbf{y}}_t$:

$$\begin{aligned}
\hat{\mathbf{y}}_t &= \lfloor \sqrt{\bar{\alpha}_t}\mathbf{y}_0 - \mathbf{u} \rceil_\Delta + \mathbf{u}, \quad \mathbf{u} \sim \mathcal{U}[{-\Delta}/2, {\Delta}/2] && (7)\\
&\overset{d}{=} \sqrt{\bar{\alpha}_t}\mathbf{y}_0 + \mathbf{u} && (8)
\end{aligned}$$

$$\begin{aligned}
\text{Var}[\mathbf{u}] &= \frac{1}{12}\left(\frac{\Delta}{2} - \frac{-\Delta}{2}\right)^2 && (9)\\
&= \frac{1}{12}\Delta^2\\
\therefore \quad \text{SNR}_{\hat{\mathbf{y}}_t} &= \frac{\mathbb{E}[(\sqrt{\bar{\alpha}_t}\mathbf{y}_0)^2]}{\frac{1}{12}\Delta^2} && (10)
\end{aligned}$$

Therefore, to match the SNR between Gaussian diffusion and our proposed method:

$$\begin{aligned}
\frac{1}{12}\Delta^2 &= 1 - \bar{\alpha}_t && (11)\\
\Delta &= \sqrt{12(1-\bar{\alpha}_t)}. && (12)
\end{aligned}$$