LOSSY IMAGE COMPRESSION WITH CONDITIONAL DIFFUSION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Denoising diffusion models have recently marked a milestone in high-quality image generation. One may thus wonder if they are suitable for neural image compression. This paper outlines an end-to-end optimized image compression framework based on a conditional diffusion model, drawing on the transform-coding paradigm. Besides the latent variables inherent to the diffusion process, this paper introduces an additional discrete "content" latent variable to condition the denoising process on. This variable is equipped with a hierarchical prior for entropy coding. The remaining "texture" latent variables characterizing the diffusion process are synthesized (either stochastically or deterministically) at decoding time. We furthermore show that the performance can be tuned toward perceptual metrics of interest. Our extensive experiments involving five datasets and 16 image perceptual quality assessment metrics show that our approach not only compares favorably in terms of rate and perceptual distortion tradeoffs but also shows robust performance under all metrics while other baselines show less consistent behavior.

1 INTRODUCTION

With visual media vastly dominating consumer internet traffic, developing new efficient codecs for images and videos has become evermore crucial (Cisco, 2017). The past few years have shown considerable progress on deep learning-based image codecs that have outperformed classical codecs in terms of the inherent tradeoff between rate (expected file size) and distortion (quality loss) (Ballé et al., 2018; Minnen et al., 2018; Minnen & Singh, 2020; Zhu et al., 2021; Yang et al., 2020; Cheng et al., 2020; Yang et al., 2022b). Recent research promises even more compression gains upon optimizing for perceptual quality, i.e., increasing the tolerance for imperceivable distortion for the benefit of lower rates (Blau & Michaeli, 2019). For example, recent works involving adversarial losses (Agustsson et al., 2019; Mentzer et al., 2020) show good perceptual quality at low bitrates.

Most state-of-the-art learned codecs currently rely on the transform coding paradigm and involve hierarchical "compressive" variational autoencoders (Ballé et al., 2018; Minnen et al., 2018; Minnen & Singh, 2020). These models simultaneously transform the data into a lower dimensional latent space and use a learned prior model for entropy-coding the latent representations into short bit strings. Using either Gaussian or Laplacian decoders, these models directly optimize for low MSE/MAE distortion performance. Given the increasing focus on perceptual performance over distortion, and given the fact that VAEs suffer from mode averaging behavior inducing blurriness (Zhao et al., 2017), one may wonder if better perceptual results can be expected by replacing the Gaussian decoder with a more expressive conditional generative model.

This paper proposes to relax the typical requirement of Gaussian (or Laplacian) decoders in compression setups and proposes a more expressive generative model instead: a conditional diffusion model. Diffusion models have achieved remarkable results on high quality image generation tasks (Ho et al., 2020; Song et al., 2021b;a). By hybridizing hierarchical compressive VAEs (Ballé et al., 2018) with conditional diffusion models, we create a novel deep generative model with promising properties for perceptual image compression. This approach is related to but distinct from the recently proposed Diff-AEs (Preechakul et al., 2022), which are neither variational (as needed for entropy coding) nor tailored to the demands of image compression.

We evaluate our new compression model on five datasets and investigate a total of 16 different metrics, ranging from distortion metrics, perceptual reference metrics, and no-reference perceptual

metrics. We find that the approach is comparable with the best available compression models while showing more consistent behavior across the different tasks. We also show that making the decoder more stochastic vs. deterministic will decrease oversmoothing while degrading distortion, showing once more that perceptual quality is distinct from good reconstruction (Blau & Michaeli, 2019).

In sum, our contributions are as follows:

- We propose the first transform-coding-based lossy compression scheme using diffusion models. The approach uses a VAE-style encoder to map images onto a content latent variable; this latent variable is then fed as context into a diffusion model for reconstructing the data. The approach can be modified to enhance several perceptual metrics of interest.
- We derive our model's loss function systematically from a variational lower bound to the data log-likelihood. The resulting distortion term is distinct from traditional VAEs and is better suited for modeling the residual noise than a conditional Gaussian distribution.
- We provided substantial empirical evidence that the approach is compatible with and, in some cases, better than the state of the art. To this end, we considered five test sets, three state-of-the-art baselines, and 16 image quality assessment metrics (classical and neural).

2 Related Work

We discuss related works on *Lossy Compression*, *Compression For Realism* and *Diffusion Models*.

Lossy Image Compression The widely-established classical codecs such as JPEG (Wallace, 1991), BPG (Bellard, 2018), WEBP (Google, 2022) have recently been challenged by end-to-end learned codecs (Ballé et al., 2018; Minnen et al., 2018; Minnen & Singh, 2020; Yang et al., 2020; Cheng et al., 2020; Zhu et al., 2021). These methods typically draw on the non-linear transform coding paradigm as realized by hierarchical VAEs. Usually, neural codecs are optimized to simultaneously minimize rate and *distortion* metrics, such as mean squared error or structural similarity.

Compression For Realism In contrast to neural compression approaches targeting traditional metrics, some recent works have explored compression models to enhance *realism* (Agustsson et al., 2019; Mentzer et al., 2020; Tschannen et al., 2018). A theoretical background for these approaches was provided by Blau & Michaeli (2019); Zhang et al. (2021), who considered optimizing the autoencoder-based compression model with additional distortion terms based on neural metrics (e.g. LPIPS (Zhang et al., 2018a)) or adversarial losses (Goodfellow et al., 2014; Rippel & Bourdev, 2017). Since GAN training introduces a variety of instabilities, successful deployment of these methods requires a variety of design choices.

Diffusion Models Probabilistic diffusion models showed impressive performance on image generation tasks, with perceptual qualities comparable to those of highly-tuned GANs while maintaining stable training (Song & Ermon, 2019; Ho et al., 2020; Song et al., 2021b; Song & Ermon, 2019; Kingma et al., 2021; Yang et al., 2022a; Ho et al., 2022; Saharia et al., 2022; Preechakul et al., 2022). Popular recent diffusion models include Dall-E2 (Ramesh et al., 2022) and Stable-Diffusion (Rombach et al., 2022). Some works also proposed diffusion models for compression. Hoogeboom et al. (2021) evaluated an autoregressive diffusion model (ADM) on a lossless compression task. Besides the difference between lossy and lossless compression, the model is only tested on low-resolution CIFAR-10 (Krizhevsky et al., 2009) dataset. In concurrent work, Theis et al. (2022) proposed a diffusion model for lossy compression, using a generic unconditional diffusion model that does not require learning a discrete representation. While conceptually attractive, the paper considers images of comparatively low resolution since "relative entropy coding" (Flamich et al., 2020) is substantially slower than transform coding (Yang et al., 2022b; Ballé et al., 2020).

3 Method

We review diffusion models and neural compression methods and then discuss our model design.

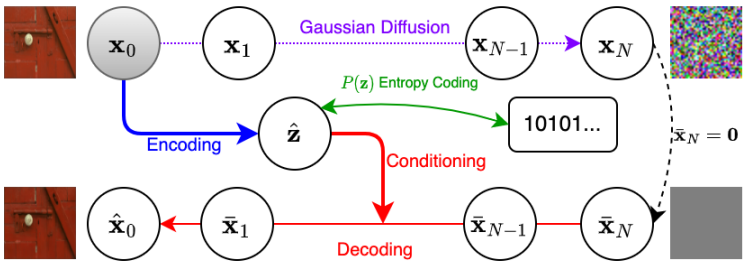


Figure 1: Overview of our proposed compression architecture. A discrete "content" latent variable $\hat{\mathbf{z}}$ contains information about the image. Upon decoding, this variable us used for conditioning a denoising diffusion process. The involved "texture" latent variables $\bar{\mathbf{x}}_{1:N}$ are synthesized on the fly.

3.1 Background

Denoising diffusion models are hierarchical latent variable models that generate data by a sequence of iterative stochastic denoising steps (Ho et al., 2020; Song et al., 2021a; Song & Ermon, 2019; Sohl-Dickstein et al., 2015). The model describes a joint distribution over data $\mathbf{x}_0$ and latent variables $\mathbf{x}_{1:N}$ such that $p_\theta(\mathbf{x}_0) = \int p_\theta(\mathbf{x}_{0:N}) d\mathbf{x}_{1:N}$. While a diffusion process (denoted by q) incrementally *destroys* structure, its reverse process p_θ *generates* structure. Both processes involve Markovian dynamics between a sequence of transitional steps (denoted by n), where

$$q(\mathbf{x}_n|\mathbf{x}_{n-1}) = \mathcal{N}(\mathbf{x}_n|\sqrt{1-\beta_n}\mathbf{x}_{n-1}, \beta_n\mathbf{I}); \quad p_\theta(\mathbf{x}_{n-1}|\mathbf{x}_n) = \mathcal{N}(\mathbf{x}_{n-1}|M_\theta(\mathbf{x}_n, n), \gamma_n\mathbf{I}). \tag{1}$$

The variance schedule $\beta_n \in (0, 1)$ can be either fixed or learned; besides it, the diffusion process is parameter-free. The denoising process predicts the posterior mean from the diffusion process and is parameterized by a neural network $M_\theta(\mathbf{x}_n, n)$. The covariance γ_n is a fixed hyperparameter.

A convenient choice to train the model is through the noise-parameterization (Ho et al., 2020), where one seeks to predict the noise used to generate a particular image:

$$L(\theta, \mathbf{x}_0) = \mathbb{E}_{n,\epsilon}||\epsilon - \epsilon_\theta(\mathbf{x}_n(\mathbf{x}_0), n)||^2. \tag{2}$$

Once the model is trained, data can be generated by following ancestral sampling similar to Langevin dynamics (Ho et al., 2020; Song & Ermon, 2019). In subsequent work, Song et al. (2021a) proposed an iterative *deterministic* mapping at training time that only injects noise in the initial draw from the prior. As we describe in Section 3.2, we adopt this deterministic scheme for image decoding.

Neural image compression seeks to outperform traditional image codecs by machine-learned models. Our approach draws on the transform-coding-based neural image compression approach (Theis et al., 2017; Ballé et al., 2018; Minnen et al., 2018; Minnen & Singh, 2020), where the data are non-linearly transformed into a latent space, and subsequently discretized and entropy-coded. The approach shows a strong formal resemblance to VAEs and shall be reviewed in this terminology.

Let $\mathbf{z}$ be a continuous latent variable and $\hat{\mathbf{z}} = \lfloor \mathbf{z} \rceil$ the corresponding rounded, integer vector. The VAE-based compression approach consists of a stochastic encoder $e(\mathbf{z}|\mathbf{x})$, a continuous prior $p(\mathbf{z})$ along with its discretization $P(\hat{\mathbf{z}})$, and a decoder $p(\mathbf{x}|\mathbf{z})$. The model is trained using the ELBO,

$$\mathcal{L}(\lambda, \mathbf{x}) = \mathbb{E}_{\mathbf{z}\sim e(\mathbf{z}|\mathbf{x})}[-\log p(\mathbf{x}|\mathbf{z}) - \lambda \log p(\mathbf{z})]. \tag{3}$$

While the first term controls the distortion, the second term controls the bitrate (upon encoding $\hat{z}$ under the prior). $e(\mathbf{z}|\mathbf{x}_0) = \mathcal{U}(\text{Enc}_\phi(\mathbf{x}_0) - \frac{1}{2}, \text{Enc}_\phi(\mathbf{x}_0) + \frac{1}{2})$ is a boxed-shaped distribution that simulates the rounding operation at training time. Once the VAE is trained, we en- and de-code data using only the deterministic components as $\hat{\mathbf{z}} = \lfloor \text{Enc}(\mathbf{x}) \rceil$ and $\hat{\mathbf{x}} = \text{Dec}(\hat{\mathbf{z}})$. We furthermore use the learned prior $P(\hat{\mathbf{z}})$ for entropy coding, e.g., using an arithmetic coder (Yang et al., 2022b).

While VAE-based approaches have used simplistic (e.g., Gaussian) decoders, we can get significantly better results when defining the decoder $p(\mathbf{x}|\mathbf{z})$ as a conditional diffusion model.

3.2 Conditional Diffusion Model for Compression

The basis of our compression approach is a new latent variable model: the diffusion variational autoencoder. This model has a "semantic" latent variable $\mathbf{z}$ for encoding the image content, and a

set of "texture" latent variables $\mathbf{x}_{1:N}$ describing residual information,

$$p(\mathbf{x}_{0:N}, \mathbf{z}) = p(\mathbf{x}_{0:N}|\mathbf{z})p(\mathbf{z}). \tag{4}$$

As detailed below, the decoder will follow a denoising process conditioned on $\mathbf{z}$. Drawing on methods described in Section 3.1, we use a neural encoder $e(\mathbf{z}|\mathbf{x}_0)$ to encode the image. The prior $p(\mathbf{z})$ is a two-level hierarchical prior (commonly used in learned image compression) and is used for entropy coding $\mathbf{z}$ after quantization (Ballé et al., 2018). Next, we discuss the novel decoder model.

Decoder We construct the conditional denoising diffusion model in a similar way to the non-variational diffusion autoencoder of Preechakul et al. (2022). Decoding $\mathbf{x}_0$ involves in the following conditional generative process similar to equation 1:

$$p_\theta(\mathbf{x}_{0:T}|\mathbf{z}) = p(\mathbf{x}_N) \prod p_\theta(\mathbf{x}_{n-1}|\mathbf{x}_n, \mathbf{z}) = p(\mathbf{x}_N) \prod \mathcal{N}(\mathbf{x}_{n-1}|M_\theta(\mathbf{x}_n, \mathbf{z}, n), \gamma_n \mathbf{I}). \tag{5}$$

Above, $\mathbf{x}_N$ is the initial texture latent variable. Our compression approach only compresses $\mathbf{z}$ and generates $\mathbf{x}_{1:N}$ at decoding time. The generative process of the model is defined by the DDIM model Song et al. (2021a), where we iterative generate/decode the latent variables $\mathbf{x}_n$ as

$$\mathbf{x}_{n-1} = \sqrt{\alpha_{n-1}} \left(\frac{\mathbf{x}_n - \sqrt{1-\alpha_n}\epsilon_\theta(\mathbf{x}_n, n, \mathbf{z})}{\sqrt{\alpha_n}} \right) + \sqrt{1-\alpha_{n-1}}\epsilon_\theta(\mathbf{x}_n, n, \mathbf{z}). \tag{6}$$

In analogy to (Ho et al., 2020), ϵ_θ is a neural network that predicts the *direction* of this iterative process and α_n is the cumulative product of the variances defined in DDIM. The crucial difference is that this process is now conditioned on $\mathbf{z}$. n denotes the step of the process.

Since most of the image content is encoded in $\mathbf{z}$, the top-level texture variable $\mathbf{x}_N$ should not contribute much information. At inference time, we therefore set $\mathbf{x}_N = \mathbf{0}$ (the point with the highest density under the prior). In our ablations, we compare against stochastic decoding, where we randomize $\mathbf{x}_N$ and/or add stochastic noise to the decoding process (with or without fixed random seed).

Optimization Objective We now derive a variational lower bound to data log-likelihood for training our model. We later discuss how we relax this bound and derive a novel rate-distortion objective.

We note that our encoder distribution $e(\mathbf{z}|\mathbf{x}_0)$ is box-shaped and has zero entropy. Defining $\lambda = 1$, our variational model minimizes a variational upper bound to the negative data log-likelihood:

$$\begin{aligned}
-\log p(\mathbf{x}_0) &\leq \mathbb{E}_{\mathbf{z}\sim e(\mathbf{z}|\mathbf{x}_0)}[-\log p(\mathbf{x}_0|\mathbf{z}) - \lambda \log p(\mathbf{z})] \\
&\leq \mathbb{E}_{\mathbf{z}\sim e(\mathbf{z}|\mathbf{x}_0)} \left[\mathbb{E}_{\mathbf{x}_{1:N}\sim q(\mathbf{x}_{1:N}|\mathbf{x}_0)} \left[-\log \frac{p(\mathbf{x}_{0:N}|\mathbf{z})}{q(\mathbf{x}_{1:N}|\mathbf{x}_0)} \right] - \lambda \log p(\mathbf{z}) \right] \qquad (7) \\
&\equiv \mathbb{E}_{\mathbf{z}\sim e(\mathbf{z}|\mathbf{x}_0)}[-\log p_{\text{lower}}(\mathbf{x}_0|\mathbf{z}) - \lambda \log p(\mathbf{z})]. \qquad (8)
\end{aligned}$$

We thereby defined $\log p_{\text{lower}}(\mathbf{x}_0|\mathbf{z}) = \mathbb{E}_{\mathbf{x}_{1:N}\sim q(\mathbf{x}_{1:N}|\mathbf{x}_0)} \left[\log \frac{p(\mathbf{x}_{0:N}|\mathbf{z})}{q(\mathbf{x}_{1:N}|\mathbf{x}_0)} \right]$ as the variational lower bound to the diffusion model's conditional data likelihood. We note that $\log p_{\text{lower}}(\mathbf{x}_0|\mathbf{z})$ generally neither has a closed-form solution, nor does it have to be a normalized probability distribution. However, the notation is still useful due to its close analogy to Gaussian decoders.

We realize that $-\log p_{\text{lower}}(\mathbf{x}_0|\mathbf{z})$ measures *image distortion*, i.e., the model's ability to reconstruct the image based on $\mathbf{z}$. In contrast, $\log p(\mathbf{z})$ measures the number of bits needed to compress $\mathbf{z}$ under the prior. By allowing λ to deviate from 1, we hence identify Eq. 7 as a generalized *rate-distortion objective* (Yang et al., 2022b). Changing λ results in different models on the rate-distortion curve.

For simplicity, we adopt the denoising score matching loss of Ho et al. (2020),

$$-\log p_{\text{lower}}(\mathbf{x}_0|\mathbf{z}) \equiv \mathbb{E}_{\mathbf{x}_0, n, \epsilon}||\epsilon - \epsilon_\theta(\mathbf{x}_n(\mathbf{x}_0), \mathbf{z}, i^n_{N_{\text{train}}})||^\ell_\ell, \quad \ell = 1 \text{ or } \ell = 2. \tag{9}$$

Instead of conditioning on n, we condition the model on the pseudo-continuous variable $i^n_N = n/N$ which yields better perceptual results and offers additional flexibility in choosing the number of denoising steps for decoding (e.g., we can use a N_{test} smaller than N_{train}). This pseudo-continuous scheme has a related continuous version (Kingma et al., 2021).

Algorithm 1 provides details on training and encoding/decoding. We find that the ℓ_1 loss leads to better perceptual qualities and shows fewer color artifacts than the ℓ_2 loss. Similar results were also reported in diffusion model for super-resolution task (Saharia et al., 2022).

Optional Perceptual Distortion While Eq. 7 already describes a viable loss function for our conditional diffusion compression model, we can influence the perceptual quality of the compressed images by introducing additional loss functions similar to (Mentzer et al., 2020).

First, we note that the decoded data point can be understood as a function of the higher-level latent $\mathbf{x}_n$, the latent code $\mathbf{z}$, and the iteration n, such that $\bar{\mathbf{x}}_0(\mathbf{x}_n, \mathbf{z}, n) = \frac{\mathbf{x}_n - \sqrt{1-\alpha_n}\epsilon_\theta(\mathbf{x}_n, \mathbf{z}, i_N^n)}{\sqrt{\alpha_n}}$. When minimizing a perceptual metric $d(\cdot, \cdot)$ in image space, we can therefore add a new term to the loss:

$$L_{\text{perceptual}} = \mathbb{E}_{\mathbf{x}_0, \epsilon, n, \mathbf{z} \sim e(\mathbf{z}|\mathbf{x}_0)}[d(\bar{\mathbf{x}}_0(\mathbf{x}_n, \mathbf{z}, n), \mathbf{x}_0)]. \tag{10}$$

$$L = \rho L_{\text{perceptual}} - (1-\rho)\mathbb{E}_{\mathbf{z} \sim e(\mathbf{z}|\mathbf{x}_0)}[\log p_{\text{lower}}(\mathbf{x}_0|\mathbf{z}) - \frac{\lambda}{1-\rho}\log p(\mathbf{z})]. \tag{11}$$

This loss term is weighted by an additional Lagrange multiplier $\rho \in [0, 1)$, resulting in a three-way tradeoff that can be analogous (but different) to Rate-Distortion-Perception trade-off (Yang et al., 2022b; Blau & Michaeli, 2019). We emphasize that our model is actually solely optimized with two special distortion terms, which both have some technical distinction to the "perception" defined by Blau & Michaeli (2019). In this paper, we choose the widely adopted LPIPS (Zhang et al., 2018a) as the perceptual loss function.

Architecture The design of the denoising module follows a similar U-Net architecture used in DDIM (Song et al., 2021a) and DDPM (Ho et al., 2020) projects. Each U-Net unit includes two ResNet blocks (He et al., 2016), one Attention block and one convolutional up/downsampling block. We use six U-Net units for both downsampling and upsampling process. The channel dimension for each downsampling unit is $64 \times j$, where j is the index of the layer range from 1 to 6; the upsampling units follow the reverse order. Each encoder module consists of one ResNet blocks and one convolutional downsampling block. For conditioning with embedding, we use ResNet blocks and transposed convolution to upscale $\mathbf{z}$ to the same spatial dimension as the inputs of the beginning four U-Net downsampling units, so that we can perform conditioning by concatenating the the output of the embedder and the input of the corresponding U-Net unit. See Appendix A and Figure 5 for more details.

Algorithm 1: Training (Left); Encoding and Decoding (Right).

while *not converged* **do**
- Sample $\mathbf{x}_0 \sim$ dataset;
- $n \sim \mathcal{U}(0, 1, 2, .., N_{train})$;
- $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$;
- $\bar{\mathbf{x}}_n = \sqrt{\alpha_n}\mathbf{x}_0 + \sqrt{1-\alpha_n}\epsilon$;
- $\hat{\mathbf{z}} \sim \mathcal{U}(\text{Enc}_\phi(\mathbf{x}_0) - \frac{1}{2}, \text{Enc}_\phi(\mathbf{x}_0) + \frac{1}{2})$;
- $\bar{\mathbf{x}}_0 = \frac{\bar{\mathbf{x}}_n - \sqrt{1-\alpha_n}\epsilon_\theta}{\sqrt{\alpha_n}}$;
- $L_{\text{D}} = |\epsilon - \epsilon_\theta(\bar{\mathbf{x}}_n, i^n_{N_{\text{train}}}\hat{\mathbf{z}})|$;
- $L = (1-\rho)L_{\text{D}} + \rho d(\bar{\mathbf{x}}_0, \mathbf{x}_0) - \lambda \log_2 P(\hat{\mathbf{z}})$;
- $(\theta, \phi) = (\theta, \phi) - \nabla_{\theta,\phi} L$

end

Given N_{test};
$\hat{\mathbf{z}} = \lfloor \text{Enc}_\phi(\mathbf{x}_0) \rceil$;
$\hat{\mathbf{z}} \overset{P(\hat{\mathbf{z}})}{\longleftrightarrow}$ binary file;
$\bar{\mathbf{x}}_N = \mathbf{0}$;
for *n*=N_{test} *to 1* **do**
- $\epsilon_\theta = \epsilon_\theta(\bar{\mathbf{x}}_n, i^n_{N_{\text{test}}}, \hat{\mathbf{z}})$;
- $\bar{\mathbf{x}}_0 = \frac{\bar{\mathbf{x}}_n - \sqrt{1-\alpha_n}\epsilon_\theta}{\sqrt{\alpha_n}}$;
- $\bar{\mathbf{x}}_{n-1} = \sqrt{\alpha_{n-1}}\bar{\mathbf{x}}_0 + \sqrt{1-\alpha_{n-1}}\epsilon_\theta$;

end
$\hat{\mathbf{x}}_0 = \bar{\mathbf{x}}_0$;
return $\hat{\mathbf{x}}_0$

4 EXPERIMENTS

We conducted a large-scale compression evaluation involving 16 image quality metrics and 5 test datasets. Besides metrics measuring differences between compressed and raw images ("full reference metrics"), we also considered "no-reference metrics" that evaluate quality without referring to any particular instance. While some of these metrics are fixed, others are learned from data. We will refer to our approach as "Conditional Diffusion Compression" (CDC) in the following.

Metrics We selected 16 metrics from multiple categories: full-reference metrics, no-reference metrics, learned metrics, and not-learned metrics. We list these metrics and their corresponding categories in Table 1. Some more recently proposed learned metrics (Zhang et al., 2018a; Prashnani

PIEAPP(Prashnani et al., 2018)	Full Reference	Learned	Perceptual Distortion
LPIPS(Zhang et al., 2018a)	Full Reference	Learned	Perceptual Distortion
DISTS(Ding et al., 2020)	Full Reference	Learned	Perceptual Distortion
CKDN(Zheng et al., 2021)	Full Reference	Learned	Perceptual Distortion
FSIM(Zhang et al., 2011)	Full Reference	Not Learned	Distortion
SSIM(Wang et al., 2004)	Full Reference	Not Learned	Distortion
MS-SSIM(Wang et al., 2003)	Full Reference	Not Learned	Distortion
CW-SSIM(Sampat et al., 2009)	Full Reference	Not Learned	Distortion
PSNR	Full Reference	Not Learned	Distortion
GMSD(Xue et al., 2013)	Full Reference	Not Learned	Distortion
NLPD(Laparra et al., 2016)	Full Reference	Not Learned	Distortion
VSI(Zhang et al., 2014)	Full Reference	Not Learned	Distortion
MAD(Larson & Chandler, 2010)	Full Reference	Not Learned	Distortion
MUSIQ(Ke et al., 2021)	No-Reference	Learned	Perception
DBCNN(Zhang et al., 2018b)	No-Reference	Learned	Perception
FID(Heusel et al., 2017)	No-Reference	Learned	Perception

Table 1: A list of the used evaluation metrics

et al., 2018; Ding et al., 2020; Zheng et al., 2021) are believed to capture perceptual similarity better than other non-learned methods. We denote perceptual full reference metrics as *perceptual distortions*. We consider FID (Heusel et al., 2017) as a no-reference metric since distances are measured on a distribution level and not per instance. For small test sets (= 100 images), we calculate FID by segmenting images into non-overlapping 256x256 resolution patches. Note that full reference metrics are considered more important than no-reference metrics since data compression ultimately amounts to transmitting information about a particular image.

Test Data To support our compression quality assessment, we consider following datasets with necessary preprocessing: **1. Kodak** (Franzen, 2013): The data consists of 24 high-quality images at 768x512 (512x768) resolution. We do not evaluate the FID score of the dataset as 24 images only yield 144 image patches. **2. Tecnick** (Asuni & Giachetti, 2014): We use 100 natural images with 600x600 resolutions. As our model currently only supports resolution (width and height) as multiples of 64px, we downsample these images to 512x512 resolution. **3. DIV2K** (Agustsson & Timofte, 2017): The validation set of this dataset contains 100 high-quality images. We resize the images with the shorter dimension being equal to 768px. Then, each image is center-cropped to a 768x768 squared shape. **4. COCO2017** (Lin et al., 2014): For this dataset, we extract all test images with resolutions higher than 512x512 and resize them to 384x384 resolution to remove compression artifacts. The resulting dataset consists of 2695 images. **5. ArtBench** (Liao et al., 2022): We use this dataset to conduct an out-of-distribution test, as it comprises 60000 images of artwork from 10 different artistic styles. We randomly select 1800 256x256 images from the *surrealism* style.

Model Training We use the **Vimeo-90k** (Xue et al., 2019) dataset to train our model, consisting of 90,000 clips of 7-frame sequences at 448x256 resolution collected from vimeo.com. This dataset is widely used for video compression research. We randomly select one frame from each clip and crop the frame randomly to 256x256 resolution in each epoch. At the beginning of training, we warm-up the model by setting $\lambda = 10^{-4}$ and keep it running for around 500,000 steps. Then, we increase λ to $\{0.0128, 0.0256, 0.0512\}$, respectively, and keep the model running for another 1,000,000 steps until the model converges. For the models with $\rho \neq 0$, we fine-tune the pretrained model with $\rho = 0$ for another 500,000 steps. We use batch_size=4 and the Adam (Kingma & Ba, 2014) optimizer in all cases. The learning rate is initialized as $lr = 5 \times 10^{-5}$ and then declines by 20% every 100,000 steps until $lr = 2 \times 10^{-5}$.

4.1 Baseline Comparisons

Baselines and Model Variants We tested two variants of our CDC model: one with $\rho = 0$ and one $\rho = 0.9$, respectively. The former version is the diffusion model with an additive perceptual reconstruction term (LPIPS). We used 500 iteration steps to decode the images, as more steps only yielded marginal improvement.

We compare our method with two state-of-the-art neural compression models: 1. **HiFiC** (Mentzer et al., 2020) is a learned compression for perceptual image compression. The model is optimized by an adversarial network and employs additional perceptual and traditional distortion losses (LPIPS

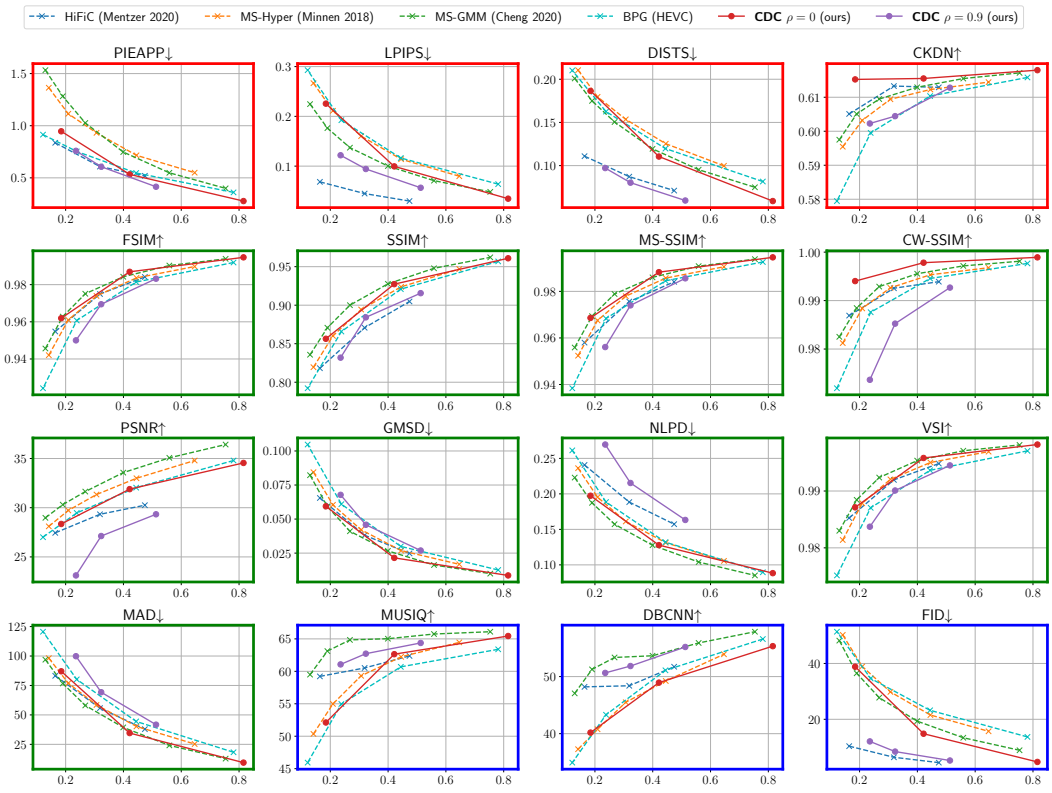


Figure 2: Tradeoffs between bitrate (x-axes, in bpp) and different perceptual metrics (y-axes) for various compression models tested on COCO2017. Arrows in the title indicate whether high (↑) or low (↓) values indicate a better perception/distortion quality. CDC (proposed) with or without finetuning to LPIPS ($\rho = 0$ or $\rho = 0.9$) shows competitive performance in various metrics.

and MSE). 2. **MS-GMM** (Cheng et al., 2020) is currently state-of-the-art in terms of rate-distortion performance (PSNR). It is based on the MSE-trained Mean-Scale Hyperprior (**MS-Hyper**) architecture (Minnen et al., 2018), but improves transform coding with an attention module and also employs an improved entropy model. 3. We also attach the results from HEVC based **BPG** codec as a reference.

Figure 2 shows the tradeoff between bitrates and perceptual metrics. Baseline models have dashed lines, and proposed ones (CDC) have solid lines. We will discuss subfigures according to their metric types, indicated by the frame color.

- **Learned full reference metrics (red frames)**. The first group includes PIEAPP, DISTS, CKDN(DR-IQA), and CKDN. We generally find that models trained with perceptual losses perform better here than models trained to minimize distortion. Our CDC($\rho = 0.9$) model shows the best PIEAPP and DISTS scores. CDC($\rho = 0$) also shows slightly better results than MS-GMM upon the above two metrics, while CDC($\rho = 0$) is slightly worse in LPIPS at low bitrates. HiFiC performs best on LPIPS as this model was optimized for this metric. For CKDN(DR-IQA), our CDC($\rho = 0$) model shows the best performance.
- **Classical rate-distorion metrics (green frames).** For the following nine traditional R-D metrics, we generally find models optimized for R-D tradeoffs to work well, such as MS-GMM (Cheng et al., 2020). CDC($\rho = 0$) shows comparable performance with MS-GMM and shows better performance over all remaining baselines. By contrast, the learned perceptual term for CDC($\rho = 0.9$) seems to rather harm performance here.
- **No-reference metrics (blue frames).** The remaining three metrics evaluate the rate-perception performance without reference to a specific image. MS-GMM shows slightly better performance on two no-reference metrics (MUSIQ, DBCNN), but our CDC($\rho = 0.9$) method is not far

(a) Ground Truth (b) CDC($\rho = 0.9$) (bpp=0.205) (c) HiFiC (bpp=0.211)

(d) Ground Truth (e) CDC($\rho = 0$) (bpp=0.228) (f) MS-GMM (bpp=0.294)

Figure 3: Qualitative comparison of the compressed images from Kodak dataset. 1^{st} row: our model more accurately retains the blue colors in the English letters and also preserve more texture pattern around the letters. HiFiC tends to sharpen the image but it comes with information loss (the color and shape of the English letters). 2^{nd} row: The low contrast texture (the curtain inside the window) is somewhat more consistent with the ground truth with our compression method.

behind. On the widely-use FID score, CDC($\rho = 0$) shows better perception over MS-GMM, and CDC($\rho = 0.9$) is only marginally worse than HiFiC.

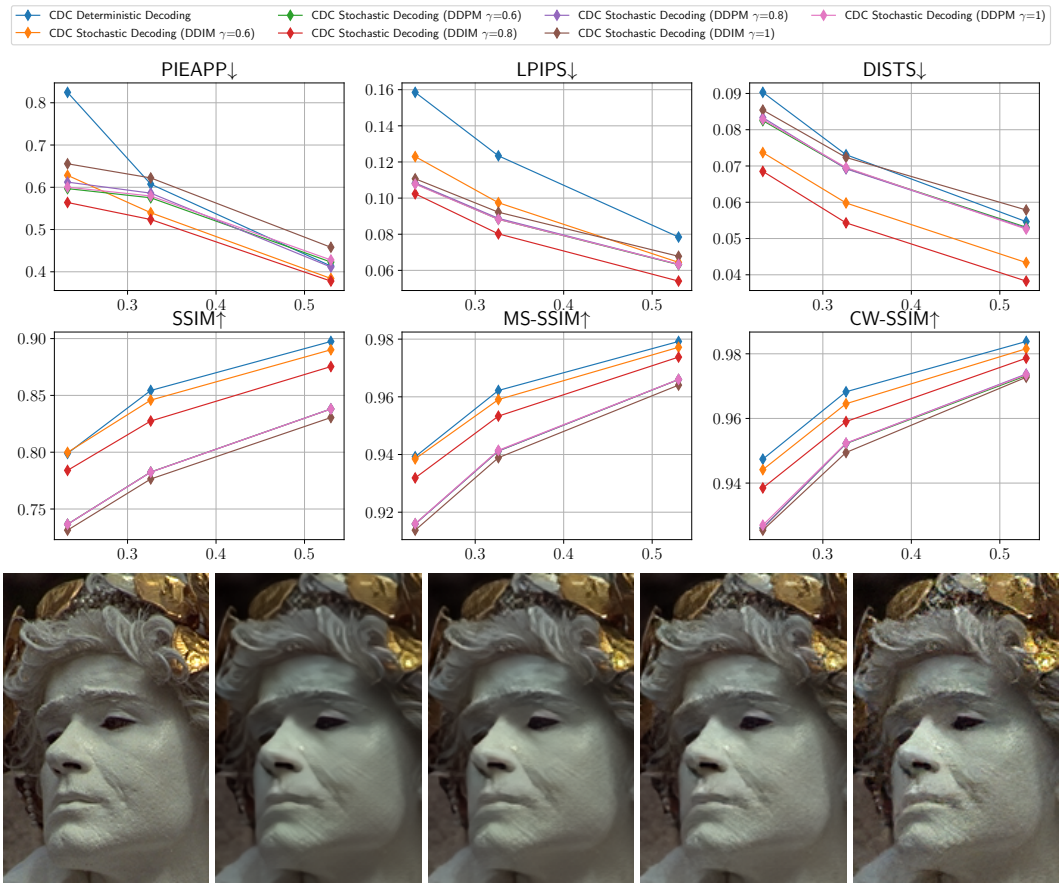
Overall, our CDC model without perceptual loss shows surprisingly good aggregate performance, despite the fact that it was not tuned towards any given metric. In contrast, the established HiFiC perceptual compression model seems to perform favorably on some perceptual metrics but is left behind on traditional metrics.

We also provide qualitative comparison of the compressed images in Figure 3. Results on all four other datasets are mostly consistent and are provided in the Appendix B, among which we show an out-of-distribution (OOD) test on ArtBench datasets (Figure 6). By benchmarking rate-distortion on these artwork data, we can report the robustness of the compression models that are optimized for natural images and perceptual quality, because perceptual quality is less important than traditional metrics for these unnatural contents. The results show that our CDC ($\rho = 0.9$) model performs better than HiFiC under such contingency.

4.2 Stochastic Decoding

Our model allows both stochastic and deterministic decoding by varying the noise level in the image generative process at decompression time. Since stochastic decoding is unintuitive and typically not desirable, one can make the decoding process still reproducible by using a fixed random seed.

To analyze the difference between stochastic and deterministic decoding, we consider both the DDIM (Song et al., 2021a) and DDPM (Ho et al., 2020) sampling schemes. DDIM sampling starts from a white noise distribution $\bar{\mathbf{x}}_0 \sim \mathcal{N}(\mathbf{0}, \gamma^2\mathbf{I})$ while making the iterative decoding process deterministic. In contrast, DDPM also involves a stochastic decoding process with noise perturbation at every decoding step. Figure 4 compares the compression performance of deterministic decoding and the two stochastic decoding schemes by evaluating three perceptual and three traditional metrics. The top plots show that DDIM and DDPM both show improved perceptual distortion performance over the deterministic variants but degrade in terms of traditional distortion metrics. By varying the noise parameter γ, DDPM shows approximately invariant scores on almost all the metrics. It also appears that the value $\gamma = 0.8$ yields the best results for DDIM in terms of both perceptual distortion and visualization quality.



(a) Ground Truth (b) Deterministic (c) DDIM $\gamma = 0.6$ (d) DDIM $\gamma = 0.8$ (e) DDIM $\gamma = 1$

Figure 4: Quantitative (top figure) and qualitative (bottom figure) comparison of deterministic and stochastic decoding methods. Deterministic decoding typically results in a smoother image reconstruction. By increasing the noise γ used upon decoding the images, we observe more and more detail and rugged texture on the face of the sculpture. Qualitatively, DDIM $(\gamma = 0.8)$ seems to show the best agreement with the ground truth image.

5 Conclusion & Discussion

This paper proposes a lossy image compression framework inspired by the conditional diffusion model and transform-coding-based neural image compression. Our approach uses a deterministic denoising decoder to iteratively reconstruct the compressed images encoded by an ordinary neural encoder. We train the model with a loss term that combines denoising score matching and rate-distortion autoencoders in an end-to-end manner. We conduct quantitative and qualitative experiments to compare our method against several state-of-the-art neural and classical codecs. Our approach yields competitive rate-distortion(perception) performance against all baseline models.

Iterative decoding can be slow compared to a normal decoder. Whether or not this is relevant depends on the application, such as the available bandwidth for transmitting data. However, we can also trade-off decoding speed against image quality by varying the decoding steps or using recent ideas for accelerating diffusion models (Salimans & Ho, 2022).

References

Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, July 2017.

Eirikur Agustsson, Michael Tschannen, Fabian Mentzer, Radu Timofte, and Luc Van Gool. Generative adversarial networks for extreme learned image compression. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 221–231, 2019.

N. Asuni and A Giachetti. Testimages: a large-scale archive for testing visual devices and basic image processing algorithms, stag - smart tools & apps for graphics conference, 2014. 2014.

Johannes Ballé, David Minnen, Saurabh Singh, Sung Jin Hwang, and Nick Johnston. Variational image compression with a scale hyperprior. *International Conference on Learning Representations*, 2018.

Johannes Ballé, Philip A Chou, David Minnen, Saurabh Singh, Nick Johnston, Eirikur Agustsson, Sung Jin Hwang, and George Toderici. Nonlinear transform coding. *IEEE Journal of Selected Topics in Signal Processing*, 15(2):339–353, 2020.

Jean Bégaint, Fabien Racapé, Simon Feltman, and Akshay Pushparaja. Compressai: a pytorch library and evaluation platform for end-to-end compression research. *arXiv preprint arXiv:2011.03029*, 2020.

Fabrice Bellard. Bpg image format. 2018. URL `https://bellard.org/bpg/`.

Yochai Blau and Tomer Michaeli. Rethinking lossy compression: The rate-distortion-perception tradeoff. In *International Conference on Machine Learning*, pp. 675–685. PMLR, 2019.

Zhengxue Cheng, Heming Sun, Masaru Takeuchi, and Jiro Katto. Learned image compression with discretized gaussian mixture likelihoods and attention modules. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7939–7948, 2020.

Cisco. Visual network index cisco. forecast and methodology. *White Paper*, 2017.

Keyan Ding, Kede Ma, Shiqi Wang, and Eero P Simoncelli. Image quality assessment: Unifying structure and texture similarity. *IEEE transactions on pattern analysis and machine intelligence*, 2020.

Gergely Flamich, Marton Havasi, and José Miguel Hernández-Lobato. Compressing images by encoding their latent representations with relative entropy coding. *Advances in Neural Information Processing Systems*, 33:16131–16141, 2020.

Richard W. Franzen. True color kodak images. 2013. URL `http://r0k.us/graphics/kodak/`.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger (eds.), *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014. URL `https://proceedings.neurips.cc/paper/2014/file/5ca3e9b122f61f8f06494c97b1afccf3-Paper.pdf`.

Google. An image format for the web; webp; google developers, 2022. URL `https://developers.google.com/speed/webp/`.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *arXiv preprint arXiv:2204.03458*, 2022.

Emiel Hoogeboom, Alexey A Gritsenko, Jasmijn Bastings, Ben Poole, Rianne van den Berg, and Tim Salimans. Autoregressive diffusion models. In *International Conference on Learning Representations*, 2021.

Junjie Ke, Qifei Wang, Yilin Wang, Peyman Milanfar, and Feng Yang. Musiq: Multi-scale image quality transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5148–5157, 2021.

Diederik Kingma, Tim Salimans, Ben Poole, and Jonathan Ho. Variational diffusion models. *Advances in neural information processing systems*, 34:21696–21707, 2021.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *IJCV*, 2020.

Valero Laparra, Johannes Ballé, Alexander Berardino, and Eero P Simoncelli. Perceptual image quality assessment using a normalized laplacian pyramid. *Electronic Imaging*, 2016(16):1–6, 2016.

Eric Cooper Larson and Damon Michael Chandler. Most apparent distortion: full-reference image quality assessment and the role of strategy. *Journal of electronic imaging*, 19(1):011006, 2010.

Peiyuan Liao, Xiuyu Li, Xihui Liu, and Kurt Keutzer. The artbench dataset: Benchmarking generative models with artworks. *arXiv preprint arXiv:2206.11404*, 2022.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.

Fabian Mentzer, George D Toderici, Michael Tschannen, and Eirikur Agustsson. High-fidelity generative image compression. *Advances in Neural Information Processing Systems*, 33:11913–11924, 2020.

David Minnen and Saurabh Singh. Channel-wise autoregressive entropy models for learned image compression. In *2020 IEEE International Conference on Image Processing (ICIP)*, pp. 3339–3343. IEEE, 2020.

David Minnen, Johannes Ballé, and George D Toderici. Joint autoregressive and hierarchical priors for learned image compression. *Advances in neural information processing systems*, 31, 2018.

Ekta Prashnani, Hong Cai, Yasamin Mostofi, and Pradeep Sen. Pieapp: Perceptual image-error assessment through pairwise preference. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1808–1817, 2018.

Konpat Preechakul, Nattanat Chatthee, Suttisak Wizadwongsa, and Supasorn Suwajanakorn. Diffusion autoencoders: Toward a meaningful and decodable representation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.

Oren Rippel and Lubomir Bourdev. Real-time adaptive image compression. In *International Conference on Machine Learning*, pp. 2922–2930. PMLR, 2017.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn 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.

Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J Fleet, and Mohammad Norouzi. Image super-resolution via iterative refinement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. *arXiv preprint arXiv:2202.00512*, 2022.

Mehul P Sampat, Zhou Wang, Shalini Gupta, Alan Conrad Bovik, and Mia K Markey. Complex wavelet structural similarity: A new image similarity index. *IEEE transactions on image processing*, 18(11):2385–2401, 2009.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pp. 2256–2265. PMLR, 2015.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021a.

Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in Neural Information Processing Systems*, 32, 2019.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021b.

Lucas Theis, Wenzhe Shi, Andrew Cunningham, and Ferenc Huszár. Lossy image compression with compressive autoencoders. *International Conference on Learning Representations*, 2017.

Lucas Theis, Tim Salimans, Matthew D Hoffman, and Fabian Mentzer. Lossy compression with gaussian diffusion. *arXiv preprint arXiv:2206.08889*, 2022.

Michael Tschannen, Eirikur Agustsson, and Mario Lucic. Deep generative models for distribution-preserving lossy compression. *Advances in neural information processing systems*, 31, 2018.

Gregory K Wallace. The jpeg still picture compression standard. *Communications of the ACM*, 34 (4):30–44, 1991.

Zhou Wang, Eero P Simoncelli, and Alan C Bovik. Multiscale structural similarity for image quality assessment. In *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, volume 2, pp. 1398–1402. Ieee, 2003.

Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

Tianfan Xue, Baian Chen, Jiajun Wu, Donglai Wei, and William T Freeman. Video enhancement with task-oriented flow. *International Journal of Computer Vision (IJCV)*, 127(8):1106–1125, 2019.

Wufeng Xue, Lei Zhang, Xuanqin Mou, and Alan C Bovik. Gradient magnitude similarity deviation: A highly efficient perceptual image quality index. *IEEE transactions on image processing*, 23(2): 684–695, 2013.

Ruihan Yang, Prakhar Srivastava, and Stephan Mandt. Diffusion probabilistic modeling for video generation. *arXiv preprint arXiv:2203.09481*, 2022a.

Yibo Yang, Robert Bamler, and Stephan Mandt. Improving inference for neural image compression. *Advances in Neural Information Processing Systems*, 33:573–584, 2020.

Yibo Yang, Stephan Mandt, and Lucas Theis. An introduction to neural data compression. *arXiv preprint arXiv:2202.06533*, 2022b.

George Zhang, Jingjing Qian, Jun Chen, and Ashish Khisti. Universal rate-distortion-perception representations for lossy compression. *Advances in Neural Information Processing Systems*, 34: 11517–11529, 2021.

Lin Zhang, Lei Zhang, Xuanqin Mou, and David Zhang. Fsim: A feature similarity index for image quality assessment. *IEEE transactions on Image Processing*, 20(8):2378–2386, 2011.

Lin Zhang, Ying Shen, and Hongyu Li. Vsi: A visual saliency-induced index for perceptual image quality assessment. *IEEE Transactions on Image Processing*, 23(10):4270–4281, 2014. doi: 10.1109/TIP.2014.2346028.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 586–595, 2018a.

Weixia Zhang, Kede Ma, Jia Yan, Dexiang Deng, and Zhou Wang. Blind image quality assessment using a deep bilinear convolutional neural network. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(1):36–47, 2018b.

Shengjia Zhao, Jiaming Song, and Stefano Ermon. Towards deeper understanding of variational autoencoding models. *arXiv preprint arXiv:1702.08658*, 2017.

Heliang Zheng, Huan Yang, Jianlong Fu, Zheng-Jun Zha, and Jiebo Luo. Learning conditional knowledge distillation for degraded-reference image quality assessment. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10242–10251, 2021.

Yinhao Zhu, Yang Yang, and Taco Cohen. Transformer-based transform coding. In *International Conference on Learning Representations*, 2021.

A ARCHITECTURES

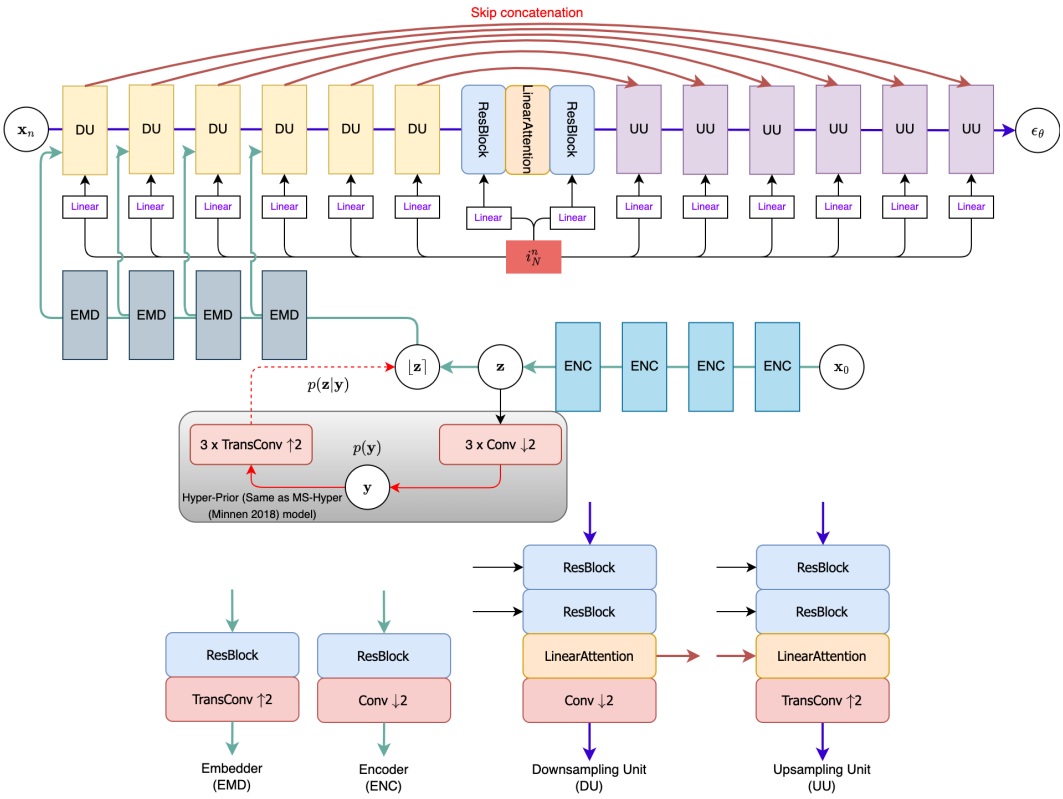


Figure 5: Visualization of our model architecture

Figure 5 describes our design choice of the model. We list the additional detailed specifications that we did not clarify in the main paper as follow:

- The hyper prior structure share the same design as Minnen et al. (2018), where the only difference is that the channel number of the hyper latent $\mathbf{y}$ is set as 256.
- We use 3x3 convolution for almost all the rest convolutional layers. The only exceptions are the 1st conv-layer of the first DU component and the 1st layer of the 1st ENC component, where we use 7x7 convolution for wider receptive field.
- i_N^n is embeded by multiple linear layers, which expand the 1-dimensional number to the sample channel size as the corresponding upsampling/downsampling units.

B ADDITIONAL RATE-DISTORTION(PERCEPTION) RESULTS

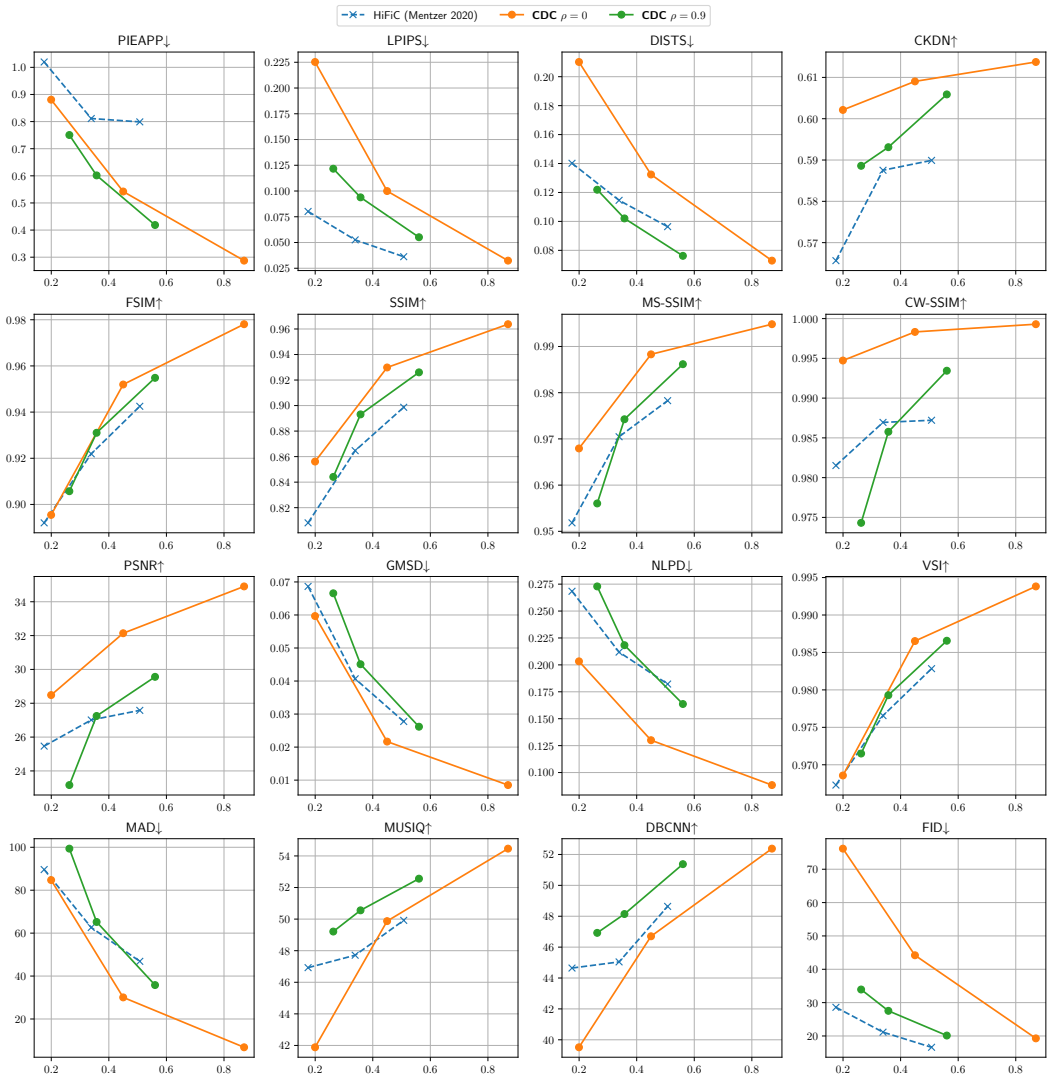


Figure 6: Rate-Distortion(Perception) for ArtBench(surrealism) dataset. This dataset, which challenges the model that prefers perceptual quality, is the only out-of-distribution dataset we conducted in this experiment. HiFiC shows worse or partially worse performance than CDC($\rho = 0.9$) in almost all of conventional distortion metrics. By contrast, in other natural image datasets (Figure 2 7 8 9 green frames) , HiFiC almost always yields better Rate-Distortion(not-learned) performance than CDC($\rho = 0.9$).

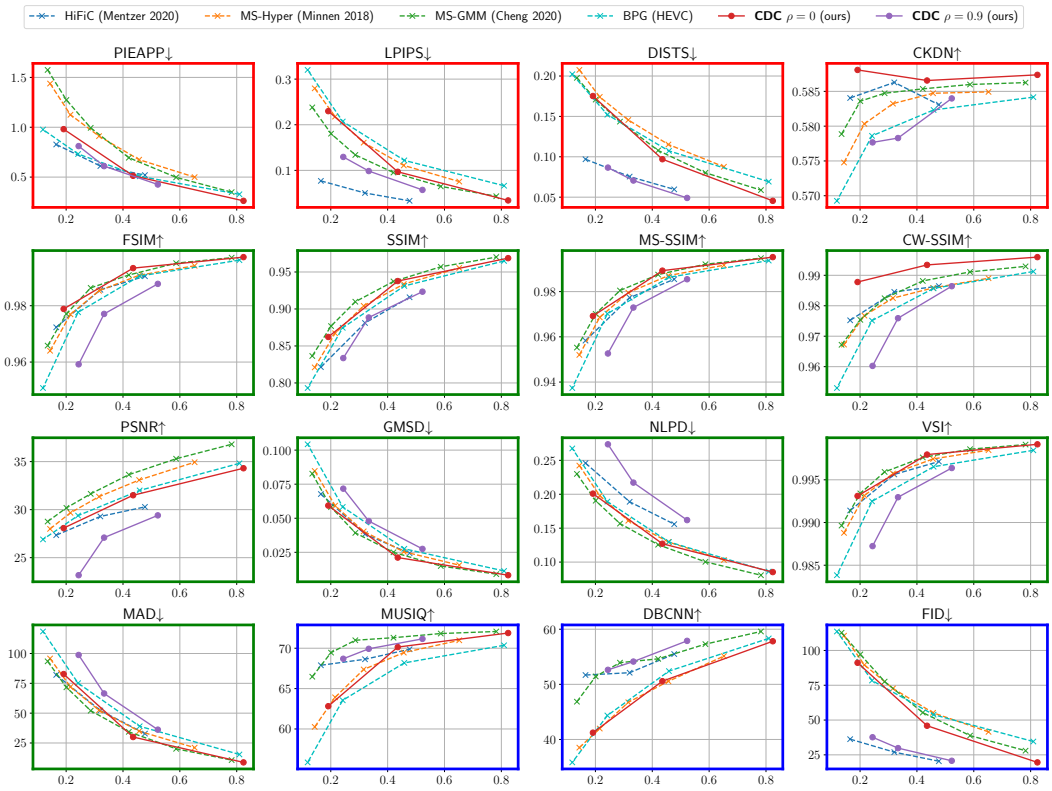


Figure 7: Rate-Distortion(Perception) for DIV2k dataset

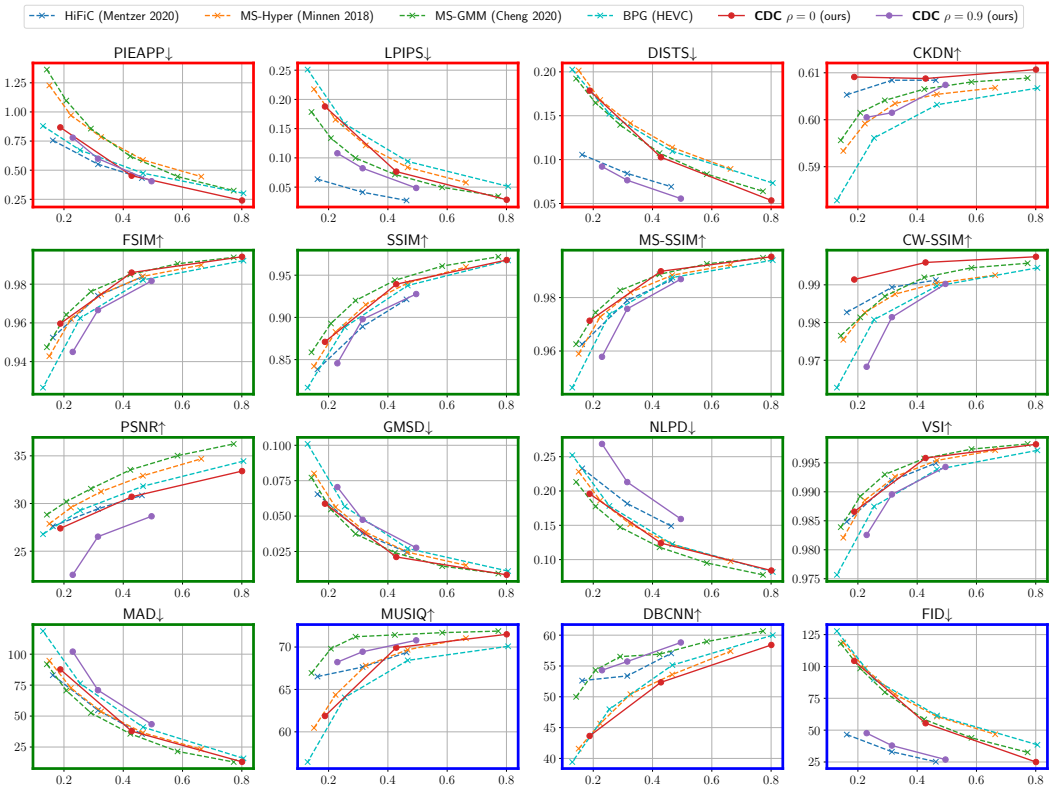


Figure 8: Rate-Distortion(Perception) for Tecnick dataset

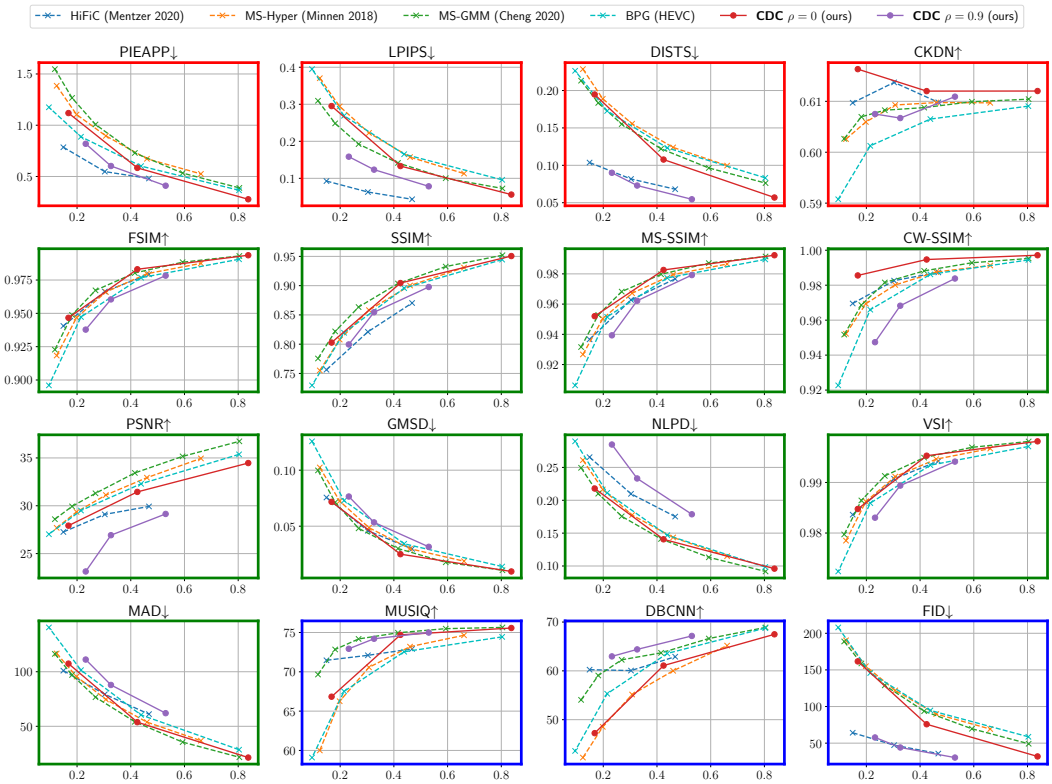


Figure 9: Rate-Distortion(Perception) for Kodak dataset

C PRETRAINED BASELINES

We refer to Bégaint et al. (2020) for pretrained MS-Hyper and MS-GMM models. For HiFiC model, we use the model implemented by a 3rd party researchers[1]. Both models were sufficiently trained on natural image datasets (Xue et al., 2019; Kuznetsova et al., 2020).

[1] https://github.com/Justin-Tan/high-fidelity-generative-compression/tree/7d4e9e785932c039df7fb436159600ed8b474a83