Controlling Rate, Distortion, and Realism: Towards a Single Comprehensive Neural Image Compression Model

Shoma Iwai   Tomo Miyazaki   Shinichiro Omachi

Graduate School of Engineering, Tohoku University, Japan

shoma.iwai.b4@dc.tohoku.ac.jp, {tomo, shinichiro.omachi.b5}@tohoku.ac.jp

Abstract

In recent years, neural network-driven image compression (NIC) has gained significant attention. Some works adopt deep generative models such as GANs and diffusion models to enhance perceptual quality (realism). A critical obstacle of these generative NIC methods is that each model is optimized for a single bit rate. Consequently, multiple models are required to compress images to different bit rates, which is impractical for real-world applications. To tackle this issue, we propose a variable-rate generative NIC model. Specifically, we explore several discriminator designs tailored for the variable-rate approach and introduce a novel adversarial loss. Moreover, by incorporating the newly proposed multi-realism technique, our method allows the users to adjust the bit rate, distortion, and realism with a single model, achieving ultra-controllability. Unlike existing variable-rate generative NIC models, our method matches or surpasses the performance of state-of-the-art single-rate generative NIC models while covering a wide range of bit rates using just one model.

1. Introduction

Image compression is a fundamental technique for efficient image storage and transmission. In recent years, neural-network-based image compression (NIC) methods have received much attention [4, 5, 23, 34]. Most NIC models are optimized to minimize the rate-distortion loss function, represented as $R + \lambda D$. R is the bit rate after compression, D is the distortion between the original image and its compressed counterpart, typically measured by mean squared error (MSE), and λ determines the balance between distortion and bit rate. State-of-the-art NIC models [23, 34] have surpassed the rate-distortion performance of the latest standard codec, Versatile Video Coding (VVC) [8].

General NIC models have two drawbacks. First, one NIC model is optimized for a specific bit rate, requiring multiple distinct models to compress an image at various bit
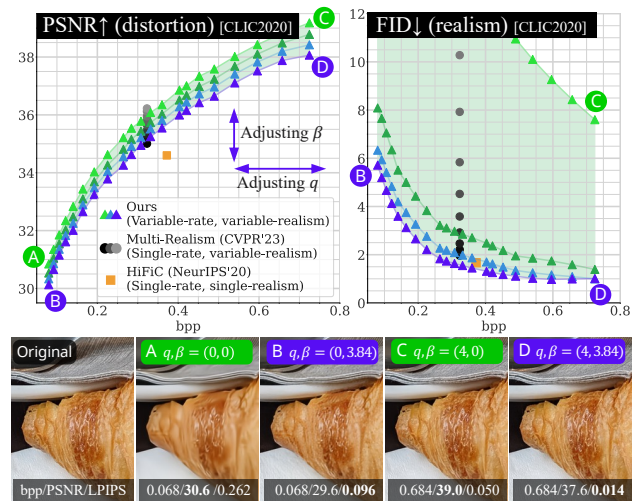

Figure 1. Top left: rate-distortion (measured by PSNR) and top right: rate-realism (measured by FID) performance using a **single model**. "bpp" stands for bits-per-pixel. While state-of-the-art GAN-based NIC methods, *Multi-Realism* [2] (capable of adjusting distortion-realism trade-off) and *HiFiC* [38] are optimized to a single bit rate, our method can control the balance between rate, distortion, and realism, covering the green area with just one model. This is achieved by adjusting two input parameters, q and β, which control the rate and the distortion-realism trade-off [6], respectively. Bottom: the original image and compression results of our method. It illustrates that our method can handle different compression settings like Ⓐ low-rate and low-distortion mode and Ⓓ high-rate and high-realism mode.

rates. Second, NIC models trained with the rate-distortion loss function tend to produce blurred images, particularly at lower bit rates, resulting in low human perceptual quality.

For the first problem, several variable-rate NIC models have been studied [14, 15, 44, 45], which can compress images at diverse bit rates using just one model. SOTA variable-rate NIC models [9, 45] demonstrate comparable performance as single-rate counterparts.

For the second problem, some works [2, 20, 38, 46] have

employed generative models such as generative adversarial networks (GAN) [21] and denoising diffusion probabilistic models (DDPM) [26] to enhance perceptual quality (realism) of reconstructions. Such generative NIC models can produce realistic images even at low bit rates.

However, only a few studies have worked on the variable-rate generative NIC model [18,20,22,36], and they come with certain limitations. Specifically, [18, 36] offers only a narrow range of adjustable bit rates, while [20, 22] suffers from poor performance compared to SOTA single-rate generative NIC models.

In this study, we propose a novel variable-rate GAN-based NIC model. Our approach differs from existing variable-rate generative NIC models in three ways. Firstly, we explore various discriminator designs tailored for the variable-rate GAN-based NIC model. Our comparison shows that the discriminator design has a substantial impact on performance. Secondly, we propose a novel adversarial loss function termed as HRRGAN (Higher Rate Relativistic GAN) to stabilize training. Thirdly, we adopt a beta-conditioning [2] to control the distortion-realism trade-off. Consequently, our method can adjust the rate-distortion-realism trade-off [7] using a single model, achieving ultra-controllability as shown in Fig 1.

Our contributions are summarized as follows:

GAN-based training tailored for variable-rate NIC model: We offer a comparison and analysis of the various discriminator designs for the variable-rate GAN-based NIC model. Moreover, we introduce a novel adversarial loss function.

High controllability and high performance: Our model can adjust the balance between rate, distortion, and realism within a single NIC model. Even with this high controllability, our model matches or outperforms the performance of state-of-the-art single-rate generative NIC models [2,38] on quantitative evaluation. To the best of our knowledge, this is the first variable-rate model that achieves the same or better performance as SOTA single-rate generative NIC models.

2. Related Works

2.1. Single Rate Neural Image Compression (NIC)

Since a VAE-based NIC model proposed by Balle et al. [4], a number of NIC models have been studied [4,5,23,41]. Typical NIC models consist of an encoder, decoder, and entropy model. Most studies focus on entropy modeling schemes to improve compression performance, such as hyperprior [5], context model [33, 40], transformer-based context model [31, 37, 42], channel autoregressive model (Charm) [41], and checkerboard context model [24]. For the encoder and decoder, attention layers [10, 13], swin-transformer-based [35] architecture [51, 52], and CNN-

Transformer mixed architecture [34] have been proposed. State-of-the-art NIC models [23, 34, 37] outperform the latest coding standard, Versatile Video Coding (VVC) [8].

2.2. Variable Rate NIC

In real-world scenarios, users adjust the compression ratio based on storage capacity or internet latency. However, most NIC models are optimized for a single rate point. Therefore, multiple models are required to compress images at different bit rates, increasing training and model storage costs. To address this issue, several variable-rate NIC methods have been studied [3,9,11,14,15,17,44,45,48]. These methods take an additional input representing the target quality and adjust the bit rate accordingly. Choi et al. [14] have introduced a conditional convolution to implement a variable-rate model. Cui et al. [15] proposed a gain-unit to modify the quantization step and control the information lost. Sun et al. [45] proposed an interpolation channel attention (ICA), allowing fine rate control without sacrificing compression performance.

2.3. Generative NIC

To improve the perceptual quality (realism) of compressed images, GANs [21] and diffusion model [26] have been incorporated into NIC. Since the pioneer work [43], several efforts have been made to improve the performance and training stability of GAN-based NIC [2,16,38]. Agustsson et al. [2] have introduced a conditional generator to control the distortion-realism trade-off [6] within a single model. Specifically, users can decide between reconstructions that have high PSNR values but appear blurry and those that look more realistic but have a lower PSNR. More recently, some diffusion-based NIC models have been proposed [20, 27, 49]. While these methods suffer from slow inference due to the iterative process, they achieve impressive perceptual quality.

However, most of the generative NIC models are single-rate models, which is not practical in real-world applications. There are only a few existing variable-rate generative NIC models that cover a wide range of bit rates. [20, 22]. Gupta et al. [22] utilize a spatial importance map to realize a variable-rate model. Another study by Ghouse et al. [20] has proposed a diffusion-based NIC model, DIRAC. It leverages a pre-trained variable-rate NIC model and enhances the reconstruction quality with a diffusion model. However, both methods are inferior to state-of-the-art single-rate models [2, 38] in terms of PSNR and FID. Moreover, DIRAC [20] requires a computationally expensive diffusion process to generate high-quality images. In this work, we bridge the performance gap between the variable-rate generative NIC model and SOTA single-rate models [2,27,38] by analyzing the discriminator design and introducing a novel adversarial loss function.
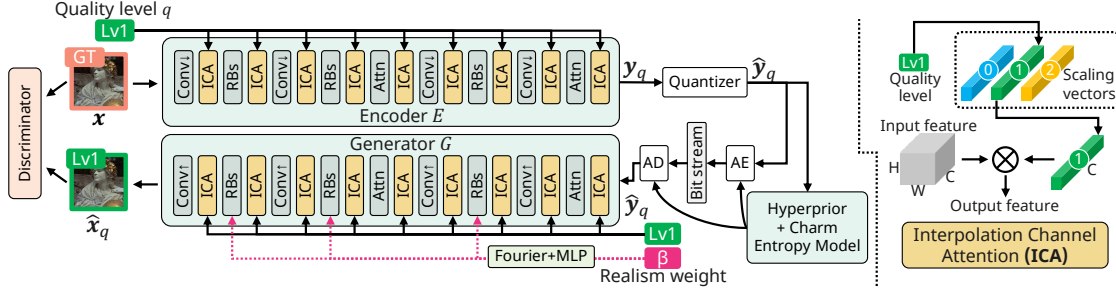
Figure 2. The overview of our NIC model. *RBs* and *Attn* in the encoder and generator stands for residual blocks and attention module used in ELIC [23]. *ICA* is an interpolation channel attention layer [45] (see right side for detail). *AE* and *AD* represent an arithmetic encoder, and arithmetic decoder, respectively. For the generator, we use beta-conditioning [2] to control the realism of the reconstruction.

3. Proposed Method

3.1. NIC pipeline

We begin by outlining the entire pipeline of our variable-rate NIC model in Fig 2. To control the rate-distortion-realism trade-off with one model, our model takes three inputs: the original image $\boldsymbol{x}$, quality level $q \in \{0, 1, \cdots, Q - 1\}$, and realism weight $\beta \in [0, \beta_{max}]$. Higher q results in a higher rate, and higher β results in higher realism. q and β are sampled randomly during training. The model is based on the state-of-the-art single-rate GAN-based NIC model, Multi-realism [2], which incorporates ELIC [23] encoder, channel autoregressive model (Charm [41]) and the beta-conditional generator [2]. We insert interpolation channel attention (ICA) layers [45] into the encoder and generator to make the model variable-rate. Each ICA layer (Fig 2 right) has Q learnable scaling vectors, and one of them is selected and applied according to q.

The compression process is as follows. First, the encoder E extracts latent representation $\boldsymbol{y}_q = E(\boldsymbol{x}, q)$. The latent $\boldsymbol{y}_q$ is then quantized into $\hat{\boldsymbol{y}}_q$ using scalar quantization. Since the quantization process is not differentiable, we use straight-through estimation (STE) during training as used in [41]. The quantized code $\hat{\boldsymbol{y}}_q$ is transformed into a bitstream losslessly by the arithmetic encoder with the probability distribution $p(\hat{\boldsymbol{y}}_q)$ estimated by the entropy model. For the details on entropy estimation, please refer to the original Charm paper [41]. The quality level q is also included in the bit stream, which occupies less than 1 byte. On the decoder side, the bitstream is first decoded back into the quantized code $\hat{\boldsymbol{y}}_q$ and quality level q with the arithmetic decoder. The generator G then reconstructs an image $\hat{\boldsymbol{x}}_q = G(\hat{\boldsymbol{y}}_q, q, \beta)$. As in [2], β is first embedded with Fourier encoding [39] and MLPs and injected into the residual blocks. Since we use GAN-based training, a discriminator is applied during the training. We describe the details of the discriminator architecture in the next section.

3.2. Exploring Discriminator

In this section, we discuss and explore various discriminator designs for variable-rate NIC model. The discriminator is trained to distinguish real (original) and fake (reconstructed) images, and the NIC model learns to reconstruct images indistinguishable from the discriminator. Although the discriminator plays an important role in GAN-based training, its design remains underexplored in current variable-rate GAN-based NIC methods [18, 22, 36].

Unlike single-rate GAN-based NIC models, variable-rate NIC models produce varying-quality images according to the quality level q. Since q is chosen randomly at every training step, the reconstruction quality can significantly vary between training steps. This behavior differs from single-rate GAN-based methods, which motivates us to design a discriminator architecture specifically tailored for the variable-rate NIC.

Fig 3 shows the various discriminator designs we explored. The base architecture of all discriminators is a CNN patch-discriminator [28] as used in [2, 38]. The output dimension is $\frac{H}{16} \times \frac{W}{16}$, where H, W are the height and width of the input image, respectively. These discriminators have two types of convolution layers: (1) those applied for all quality levels and (2) those applied for a specific quality level. For layer (1), we introduce a quality condition to provide the discriminator information about the quality level. Specifically, given q, we employ one-hot encoding and expand this to get a conditional feature $C \in \{0, 1\}^{h \times w \times Q}$. C is concatenated with the input of the discriminator's first convolution layer (1). While layer (1) can capture features common across all quality levels (inherent in the NIC model), layer (2) can learn unique features on the individual quality level. We aim to find which features are more beneficial for the NIC model by comparing different designs. We describe the details of each design as follows.

(a) Independent discriminator consists of Q distinct sub-discriminators. Each sub-discriminator is applied to reconstructions of one particular quality level. However,
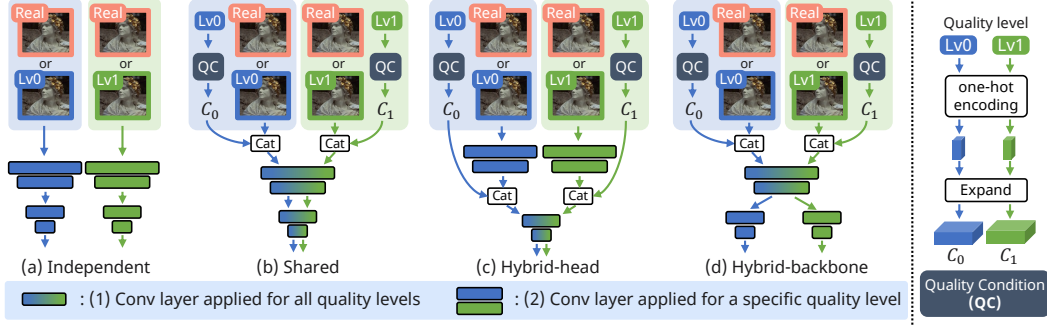
Figure 3. The discriminator designs that we consider. Discriminators take an original image or its reconstruction as input and estimate the reality of the input. They consist of two kind of convolution layers: (1) layers applied for all quality levels and (2) layers applied for a specific quality level. For layer (1), we introduce a quality condition (right).
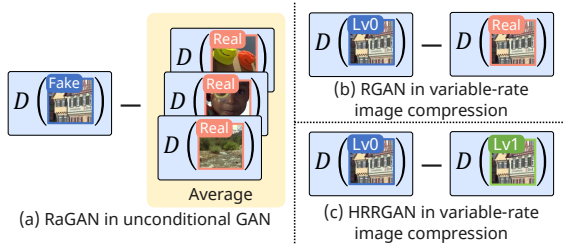

Figure 4. How to calculate the relative "reality score" on (a) Ra-GAN in unconditional GAN, (b) RGAN in variable-rate image compression, and (c) HRRGAN in variable-rate image compression.

it cannot leverage features common across various quality levels.

(b) Shared discriminator is a single CNN discriminator applied across all quality levels. While it can capture the features ubiquitous across all quality levels, it cannot learn the quality-level-specific features.

(c), (d) Hybrid-head and -backbone discriminator have both types of layers to leverage quality-level-specific and common features. (c) has independent backbones and a shared prediction head, whereas (d) has a shared backbone and independent prediction heads. By comparing them, we aim to analyze the results when each layer type is applied to low- and high-level features.

Based on the comparison of performance (see Fig 7), we use *(a) independent discriminator*. We will discuss the detailed results in Sec.4.4. Note that the discriminator is used only in the training; thus, using different layers for each quality level does not affect the inference process, such as model size and encoding/decoding speed.

3.3. Higher Rate Relativistic GAN (HRRGAN)

In this section, we describe our novel adversarial loss function, Higher Rate Relativistic GAN (HRRGAN), which

is inspired by Relativistic average GAN (RaGAN) [29]. In some generative NIC methods [12, 18], RaGAN has been used to improve perceptual quality. In RaGAN, the adversarial losses for the generator and discriminator are calculated as follows:

$$p_r(x_r, x_f) = \sigma(D(x_r) - \mathbb{E}_{x_f}[D(x_f)])$$
$$p_f(x_r, x_f) = \sigma(D(x_f) - \mathbb{E}_{x_r}[D(x_r)])$$
$$\mathcal{L}_{\text{RaGAN}}^G = -\log p_f(x_r, x_f) - \log(1 - p_r(x_r, x_f)) \quad (1)$$
$$\mathcal{L}_{\text{RaGAN}}^D = -\log p_r(x_r, x_f) - \log(1 - p_f(x_r, x_f)), \quad (2)$$

where D is the discriminator, and x_r, x_f are real and fake images, respectively. $\sigma(\cdot)$ and $\mathbb{E}_{x_f}[\cdot]$ represent sigmoid function and the average operation for all x_f in the mini batch, respectively. Intuitively, in RaGAN, the discriminator D estimates the "reality score" of an input image instead of the probability that the input image is real as the standard GAN [21]. Then, the generator is trained so that the reality score of the fake image are higher than those of real images on average.

Our HRRGAN is different from RaGAN in two ways. First, we omit the average computation in Eq 1, which is the same loss function as the Relativistic standard GAN (RGAN) [29]. In unconditional GAN, the average computation is necessary because x_r and x_f are not aligned as shown in Fig 4(a). However, in the image compression task, the fake image is a reconstructed version of the original, ensuring that the x_r and x_f are spatially aligned (Fig 4(b)). To leverage this alignment and calculate the relative reality score for each region, we do not average the discriminator's outputs as follows:

$$\mathcal{L}_{\text{RGAN}}^G = -\log \sigma(D(x_f) - D(x_r)) \quad (3)$$
$$\mathcal{L}_{\text{RGAN}}^D = -\log \sigma(D(x_r) - D(x_f)). \quad (4)$$

Second, we replace the real image with a reconstruction of higher quality when training the NIC model in Eq 4. In
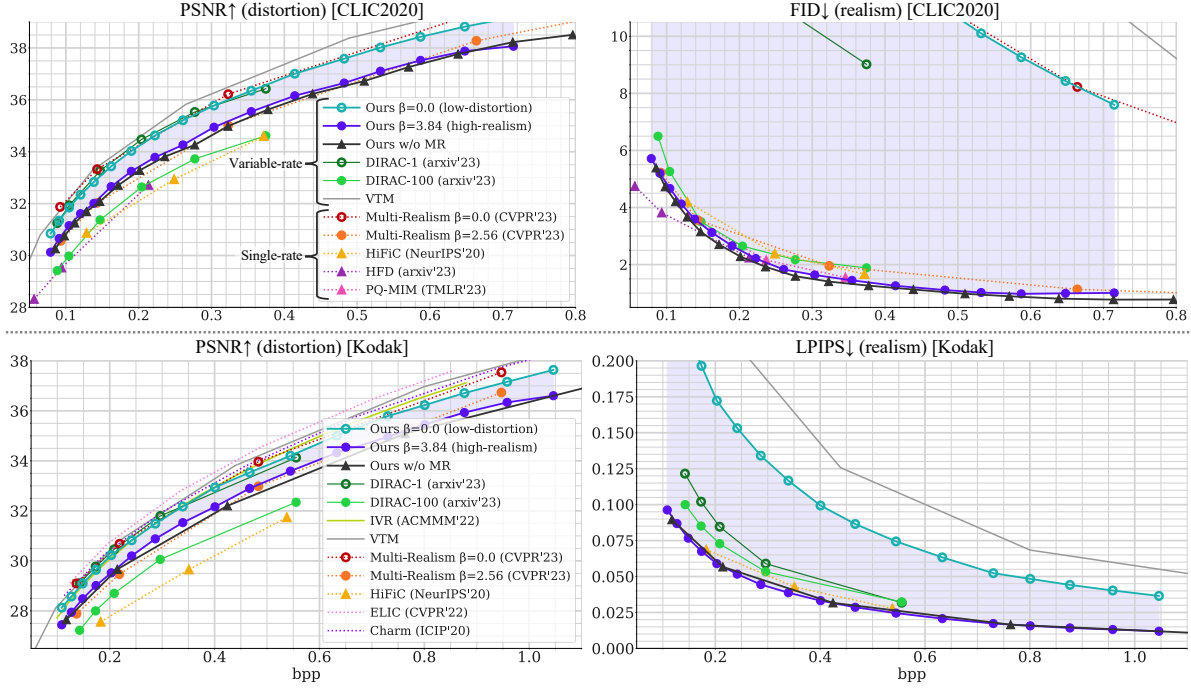
Figure 5. Quantitative results on CLIC2020 test (top) and Kodak dataset (bottom). We use PSNR to evaluate the rate-distortion performance and FID and LPIPS for the rate-realism performance. Solid lines represent variable-rate methods, while dashed lines denote single-rate methods. As for the markers, circles (● and ○) represent variable-realism NIC, triangles (▲) indicate generative NIC, and lines without markers indicate non-generative methods. We report LPIPS on CLIC2020 dataset in the supplementary material.

RGAN, the compression model is trained so that the reconstruction is estimated as more realistic than the original. We found that this approach imposes an over-penalty on the NIC model, resulting in excessive loss values even for successful reconstructions. To mitigate this problem, when the model reconstructs image $\hat{\boldsymbol{x}}_q$ with a quality level q, we use another reconstruction $\hat{\boldsymbol{x}}_{q+1}$ with quality level $q + 1$ to calculate the relative reality score. The loss function of our HRRGAN is represented as follows:

$$\mathcal{L}_{\text{HRRGAN}}^{G} = -\log \sigma(D(\hat{\boldsymbol{x}}_q) - \text{sg}(D(\hat{\boldsymbol{x}}_{q+1}))) \quad (5)$$

$$\mathcal{L}_{\text{HRRGAN}}^{D} = -\log \sigma(D(\boldsymbol{x}) - D(\hat{\boldsymbol{x}}_q)), \quad (6)$$

where sg denotes the stop gradient operation, preventing the NIC model from generating less realistic $\hat{\boldsymbol{x}}_{q+1}$ to minimize Eq 5. Although $\hat{\boldsymbol{x}}_{q+1}$ has higher quality than $\hat{\boldsymbol{x}}_q$, its quality is not high as $\boldsymbol{x}$. As a result, HRRGAN relaxes the loss function for the NIC model and reduces the chance of over-penalizing, leading to more balanced training. Note that the same loss function as RGAN is used for the discriminator.

3.4. Training

Following [38], our training consists of two stages. In the first stage, we train the model without adversarial loss

using the following loss function:

$$\mathcal{L}_{1st} = \lambda_R^{(q)} R(\hat{\boldsymbol{y}}_q) + \lambda_d d(\boldsymbol{x}, \hat{\boldsymbol{x}}_q) + \mathcal{L}_P(\boldsymbol{x}, \hat{\boldsymbol{x}}_q), \quad (7)$$

where $R, d, \mathcal{L}_P$ represent the bit rate estimated by the entropy model, MSE, and LPIPS [50], respectively. The weight of the bit rate $\lambda_R^{(q)} \in \{\lambda_R^{(0)}, \lambda_R^{(1)}, \cdots, \lambda_R^{(Q-1)}\}$ is selected according to the quality level.

In the second stage, we fine-tune the model with the proposed adversarial loss as follows:

$$\mathcal{L}_{2nd} = \lambda_R^{(q)} R(\hat{\boldsymbol{y}}_q) + \lambda_d d(\boldsymbol{x}, \hat{\boldsymbol{x}}_q)$$
$$+ \beta(\lambda_P \mathcal{L}_P(\boldsymbol{x}, \hat{\boldsymbol{x}}_q) + \lambda_{\text{adv}} \mathcal{L}_{\text{HRRGAN}}^{G}), \quad (8)$$

where β balances the influence of LPIPS and adversarial loss. As a result, the NIC model learns to generate high-realism images with a higher value of input β.

4. Experiments

4.1. Experimental settings

Dataset. In our experiment, we trained our model on the subset of OpenImage dataset [32], which contains about 1M images. We randomly crop the images into 256×256 patches and apply random horizontal flipping. The batch size is set to 8. We evaluate models on standard benchmarks
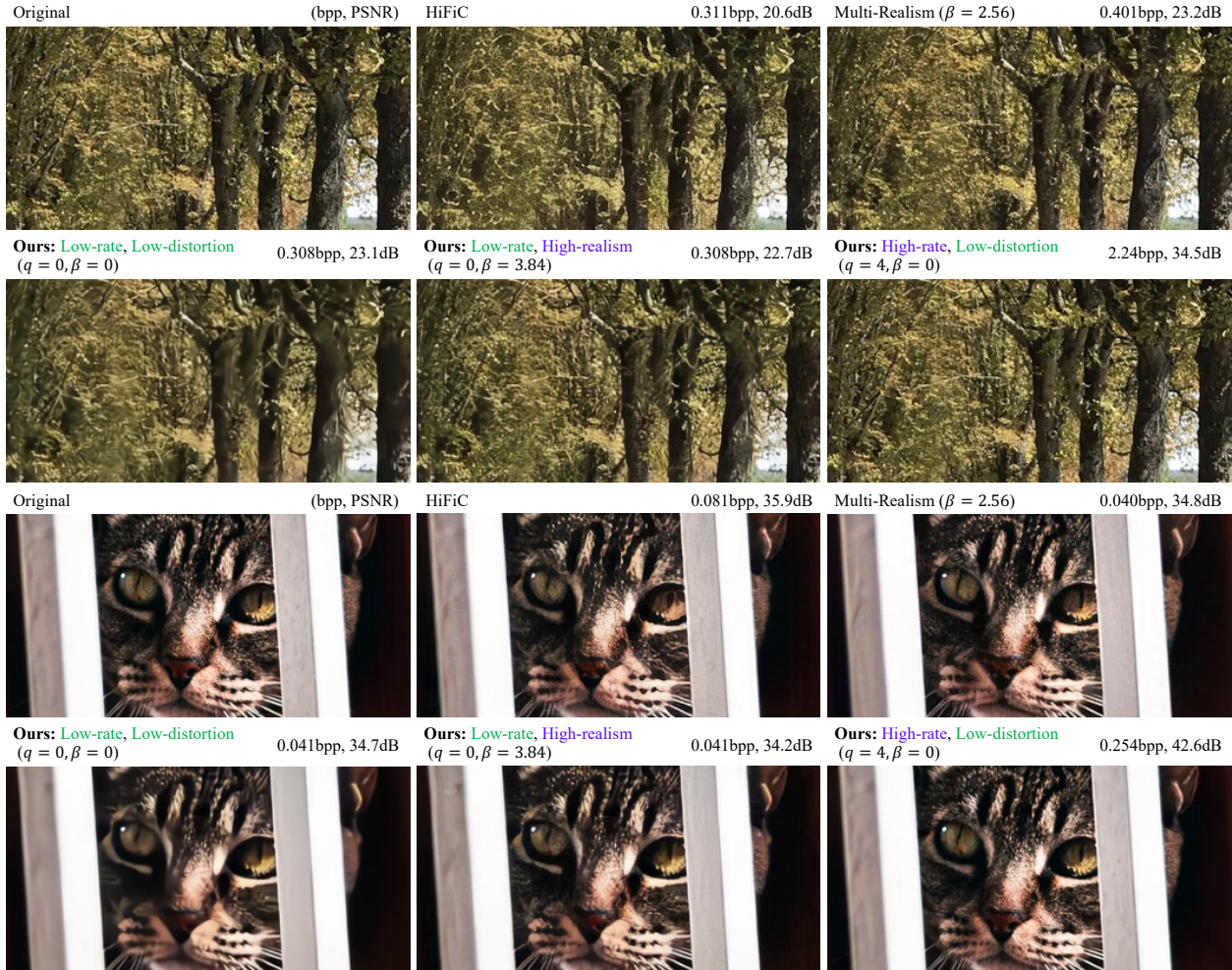
Figure 6. Comparing the original image to reconstructions of our model and state-of-the-art generative NIC models, HiFiC [38] and Multi-Realism [2] on CLIC2020 dataset. We show our three reconstructions with different configurations: $(q, \beta) = (0, 0), (0, 3.84), (4, 0)$.

for the image compression task, Kodak [1] (24 images) and CLIC2020 test dataset [47] (428 images).

Evaluation. We use PSNR for distortion (fidelity) performance and Frechét Inception Distance (FID) [25] and Learned Perceptual Image Patch Similarity (LPIPS) [50] for realism (perceptual quality) performance. We followed the protocol in [38] to calculate FID. We did not use FID for Kodak because it contains only 24 images.

Training settings. We optimize the model using Adam [30]. We set the training steps of each training stage to $2M$ and $3M$. In each stage, the initial learning rate is set to 1.0×10^{-4} and decayed to 1.0×10^{-5} for the last 20% of the total iterations. The same learning rate settings are applied to the discriminator. The number of quality levels is $Q = 5$, and β_{max} is set to 5.12 as in [2]. For the loss function, we set $\{\lambda_R^{(0)}, \cdots, \lambda_R^{(4)}\} = \{3.4, 1.3, 0.4, 0.12, 0.05\}$, $\lambda_d = 150$, $\lambda_P = 2/\beta_{max}$, and $\lambda_{\text{adv}} = 0.002/\beta_{max}$.

Baselines. We compare our variable-rate GAN-based NIC model with existing compression methods. These methods are divided into three groups: generative NIC models, non-generative NIC models, and non-learning-based codec. Generative NIC models include GAN-based methods (Multi-realism [2], HiFiC [38], and PQ-MIM [16]) and diffusion-based methods (HFD [27] and DIRAC [20]). Non-generative NIC models, which are optimized for PSNR, include ELIC [23], Charm [41], and IVR [45]. As a non-learning-based codec, we use the latest codec, VTM [19] (software based on VVC [8]). Except for DIRAC, IVR, and VTM, all baselines are single-rate methods, requiring separate models for different bit rates. Multi-realism [2] and DIRAC [20] can control the distortion-realism trade-off with one model. Specifically, *Multi-realism* $\beta = 0$ and *DIRAC-1* represent low-distortion mode, while *Multi-realism* $\beta = 2.56$ and *DIRAC-100* are high-realism mode.
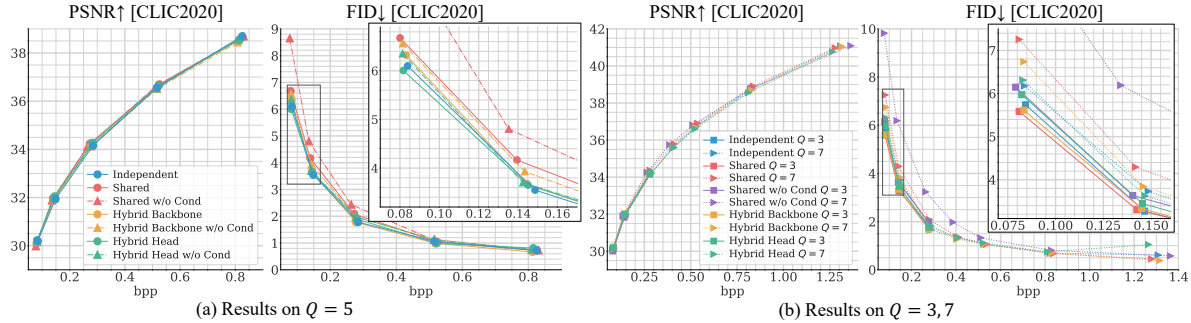
Figure 7. Quantitative comparison among different discriminator designs. *w/o Cond* indicates that the discriminator does not take the condition C as input. (a) shows the results on $Q = 5$. (b) shows the results on $Q = 3$ and $Q = 7$

4.2. Quantitative Comparison

Figure 5 illustrates quantitative results on CLIC2020 and Kodak datasets at different bit rates. Although our model is trained on Q quality levels, we perform fine rate-tuning during inference by interpolating the scaling vectors [45], obtaining reconstructions on 17 different rate points. For *Ours*, we show results on low-distortion mode ($\beta = 0$) and high-realism mode ($\beta = 3.84$). We report results on other β in the supplementary material. *Ours w/o MR* is the model trained with the fixed realism weight, $\beta = 2.56$.

First, we compare our high-realism mode (*Ours ($\beta = 3.84$)*) with other generative NIC models. On CLIC2020 dataset, *Ours ($\beta = 3.84$)* surpassed the variable-rate generative model, DIRAC-100, on both PSNR and FID. Compared to the SOTA single-rate model, Multi-Realism ($\beta = 2.56$) [2], our method demonstrates superior FID and competitive PSNR, despite using a single model for different rates. Although the recent diffusion-based method, HFD [27], achieves the best FID at low rate points, we significantly excel in PSNR. On Kodak dataset, we outperform DIRAC-100 [20] and HiFiC [38] on both PSNR and LPIPS. Furthermore, on both datasets, *Ours ($\beta = 3.84$)* achieves comparable performance as *Ours w/o MR*, indicating that the β-conditioning [2] does not hurt the realism performance even in variable-rate model.

Our low-distortion mode ($\beta = 0.0$) achieves comparable performance as other low-distortion mode models, *Multi-Realism ($\beta = 0.0$)* and *DIRAC-1*, in terms of PSNR on CLIC2020. Moreover, the PSNR values of *Ours ($\beta = 0.0$)* on Kodak match other distortion-oriented variable-rate models (*DIRAC-1* and *IVR*). These results suggest that both modes of our model achieve strong performance while covering a wide range of rates.

4.3. Qualitative Comparison

We present qualitative comparisons with state-of-the-art generative NIC models [2, 38] in Fig 6. In the first example, our $(q, \beta) = (0, 3.84)$ reconstruction preserves the texture

of leaves more effectively than HiFiC [38]. Furthermore, it has comparable perceptual quality as Multi-Realism [2]. In the second sample, while HiFiC, Multi-Realism, and our $(q, \beta) = (0, 3.84)$ mode display similar levels of realism, HiFiC uses a double bit rate. Comparing our three reconstructions, our $(q, \beta) = (0, 0)$ reconstructions appear blurry but have higher PSNR than $(q, \beta) = (0, 3.84)$ counterpart, confirming that our model controls the distortion-realism trade-off. Moreover, although our $(q, \beta) = (4, 0)$ reconstructions use high bit rates, they preserve contents faithfully and achieve higher PSNR. In conclusion, the qualitative results demonstrate that our method works well for different use cases with only one model while matching the visual performance of state-of-the-art methods.

4.4. Impact of Discriminator Design

In this section, we analyze the impact of discriminator designs. We used the NIC models without Charm [41] and trained them with fixed $\beta = 2.56$ to save computation costs.

Fig 7(a) compares the rate-distortion-realism performance on different designs on $Q = 5$. *w/o Cond* indicates that a quality condition C is not used. Through the comparison, we make the following observations. First, the discriminator designs have a significant impact on FID, especially at low bit rates. However, there is no significant difference in terms of PSNR. Second, the information about the quality level is crucial. In Fig 7(a), only the *Shared w/o Cond* takes no information about the quality level (i.e., no level-specific layer and quality level condition) and results in clearly the worst FID. Third, the level-specific layer is beneficial. *Shared* does not have a level-specific layer and performs worse than other designs with level-specific layers, suggesting the necessity of the quality-level-specific layers. Finally, convolution layers applied for all quality levels are not necessary. While *Independent* does not have a layer shared across all levels, the performance is comparable to or better than hybrid designs: *Hybrid Backbone* and *Hybrid Head*. It indicates that capturing features com-
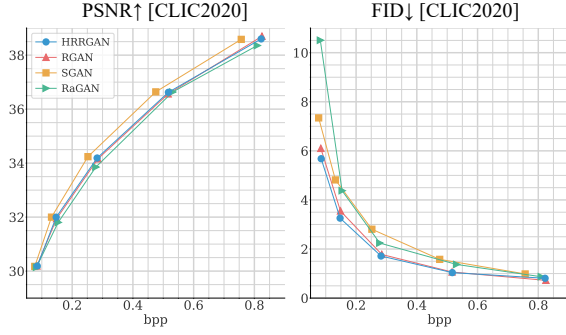
Figure 8. Results on different adversarial loss functions: HRRGAN (ours), Relativistic GAN (RGAN), standard GAN (SGAN) [21], and Relativistic average GAN (RaGAN) [29].
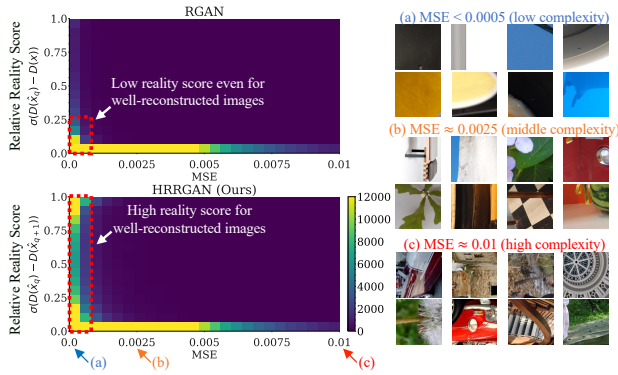

Figure 9. The two-dimensional histogram shows the relationship between reconstruction MSE and relative "reality score" estimated by the discriminator on RGAN and HRRGAN. The lower the reality score is, the higher the adversarial loss becomes. The right side shows examples of training samples with different MSE values. It suggests that the samples with low MSE (a) have lower complexity than those with higher MSE (b) and (c).

mon across all quality levels may not benefit the NIC model, whether at shallow or deep layers.

To further examine the impact of discriminators, we trained the models with wider and narrower bit rate ranges. Specifically, we present the results on $Q = 7$ (roughly 0.08 $\sim$ 1.3bpp) and $Q = 3$ (roughly 0.08 $\sim$ 0.3bpp) in Fig 7(b). For $Q = 7$, we observed a similar trend to Fig 7(a), where *Shared* and *Shared w/o Cond* result in clearly high FID. On the other hand, for $Q = 3$, *Shared* performs comparably as *Independent*, and the performance gap between *Shared w/o Cond* and other designs is less substantial than in $Q = 5, 7$. These results indicate that the discriminator design is particularly crucial for the NIC model with a wider bit rate range. In addition, Fig 7(b) demonstartes that *Independent* performs robustly on different Q.

4.5. Effect of HRRGAN

To verify the effectiveness of our HRRGAN, we compare it with other adversarial loss functions: standard GAN (SGAN) [21], Relativistic GAN (RGAN), and Relativistic average GAN (RaGAN) [29]. We used NIC models without Charm [41] and trained them with fixed $\beta = 2.56$ to save computation costs. Fig 8 shows the results on CLIC2020. Regarding FID, HRRGAN performs best, particularly at lower bit rates. As discussed in Sec 3.3, HRRGAN relaxes the loss function of RGAN, leading to more balanced training and better performance. Although SGAN achieves the highest PSNR at middle and high rates, it performs worse on FID. Moreover, RGAN consistently surpasses RaGAN, indicating that the average calculation of RaGAN harms performance in the image compression task.

To further analyze the effect of HRRGAN, we show a 2D histogram representing the relationship between the reconstruction MSE and relative "reality score" of RGAN and HRRGAN in Fig 9. For this analysis, we generated 4000 reconstructions with dimensions of 256×256 from OpenImage validation dataset. These reconstructions are fed into the trained discriminator. Then, we calculate MSE and the relative reality score: $D(\hat{x}_q) - D(x)$ for RGAN, and $D(\hat{x}_q) - D(\hat{x}_{q+1})$ for HRRGAN. The histogram shows that RGAN assigns low reality scores (i.e., high adversarial loss) even to samples with extremely low MSE like Fig 9 (a). This leads to an excessive penalty for easy and well-reconstructed samples. In contrast, HRRGAN tends to output high reality scores (i.e., low adversarial loss) for low MSE samples. It mitigates the risk of over-penalty and encourages the NIC model to focus on refining complex and challenging samples like Fig 9 (b) and (c).

5. Conclusion

We have proposed a novel NIC model that can control the rate-distortion-realism trade-off [7] with one model by adjusting two input parameters, q and β. We have tried various discriminator designs and found that quality-level-specific layers are important for variable-rate generative NIC. Moreover, inspired by RaGAN [29], we proposed HRRGAN to avoid over-penalty. In the experiments, our method achieved state-of-the-art rate-distortion-realism performance with one NIC model.

For limitations, although our method realizes high controllability, q and β control rate and realism uniformly. Consequently, it cannot perform pixel-level control. In real-world applications, however, precisely preserving specific regions (e.g., small faces) is important. Therefore, integrating pixel-level control as in [17,44] will be our future work.

References

[1] Kodak photodc dataset, 1991. https://r0k.us/graphics/kodak/. 6

[2] Eirikur Agustsson, David Minnen, George Toderici, and Fabian Mentzer. Multi-realism image compression with a conditional generator. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2023. 1, 2, 3, 6, 7

[3] Federico Baldassarre, Alaaeldin El-Nouby, and Hervé Jégou. Variable rate allocation for vector-quantized autoencoders. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023. 2

[4] Johannes Ballé, Valero Laparra, and Eero P. Simoncelli. End-to-end optimized image compression. In *International Conference on Learning Representations (ICLR)*, 2017. 1, 2

[5] Johannes Ballé, David Minnen, Saurabh Singh, Sung Jin Hwang, and Nick Johnston. Variational image compression with a scale hyperprior. In *International Conference on Learning Representations (ICLR)*, 2018. 1, 2

[6] Yochai Blau and Tomer Michaeli. The perception-distortion tradeoff. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 1, 2

[7] Yochai Blau and Tomer Michaeli. Rethinking lossy compression: The rate-distortion-perception tradeoff. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, 2019. 2, 8

[8] Benjamin Bross, Ye-Kui Wang, Yan Ye, Shan Liu, Jianle Chen, Gary J. Sullivan, and Jens-Rainer Ohm. Overview of the versatile video coding (vvc) standard and its applications. *IEEE Transactions on Circuits and Systems for Video Technology*, 31(10):3736–3764, 2021. 1, 2, 6

[9] Shilv Cai, Zhijun Zhang, Liqun Chen, Luxin Yan, Sheng Zhong, and Xu Zou. High-fidelity variable-rate image compression via invertible activation transformation. In *Proceedings of the 30th ACM International Conference on Multimedia (ACMMM)*, 2022. 1, 2

[10] Tong Chen, Haojie Liu, Zhan Ma, Qiu Shen, Xun Cao, and Yao Wang. End-to-end learnt image compression via non-local attention optimization and improved context modeling. *IEEE Transactions on Image Processing*, 30:3179–3191, 2021. 2

[11] Tong Chen and Zhan Ma. Variable bitrate image compression with quality scaling factors. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020. 2

[12] Zhengxue Cheng, Ting Fu, Jiapeng Hu, Li Guo, Shihao Wang, Xiongxin Zhao, Dajiang Zhou, and Yang Song. Perceptual image compression using relativistic average least squares gans. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2021. 4

[13] Zhengxue Cheng, Heming Sun, Masaru Takeuchi, and Jiro Katto. Learned image compression with discretized gaussian mixture likelihoods and attention modules. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[14] Yoojin Choi, Mostafa El-Khamy, and Jungwon Lee. Variable rate deep image compression with a conditional autoencoder.

[15] Ze Cui, Jing Wang, Shangyin Gao, Tiansheng Guo, Yihui Feng, and Bo Bai. Asymmetric gained deep image compression with continuous rate adaptation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 1, 2

[16] Alaaeldin El-Nouby, Matthew J. Muckley, Karen Ullrich, Ivan Laptev, Jakob Verbeek, and Herve Jegou. Image compression with product quantized masked image modeling. *Transactions on Machine Learning Research*, 2023. 2, 6

[17] Chenjian Gao, Tongda Xu, Dailan He, Hongwei Qin, and Yan Wang. Flexible neural image compression via code editing. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. 2, 8

[18] Shangyin Gao, Yibo Shi, Tiansheng Guo, Zhongying Qiu, Yunying Ge, Ze Cui, Yihui Feng, Jing Wang, and Bo Bai. Perceptual learned image compression with continuous rate adaptation. In *4th Challenge on Learned Image Compression (CLIC)*, 2021. 2, 3, 4

[19] Fraunhofer Gesellschaft. VTM-17.1, 2022. https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware_VTM/-/releases/VTM-17.1. 6

[20] Noor Fathima Ghouse, Jens Petersen, Auke Wiggers, Tianlin Xu, and Guillaume Sautière. A residual diffusion model for high perceptual quality codec augmentation. arXiv preprint arXiv:2301.05489, 2023. 1, 2, 6, 7

[21] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *International Conference on Neural Information Processing Systems*, 2014. 2, 4, 8

[22] Rushil Gupta, Suryateja BV, Nikhil Kapoor, Rajat Jaiswal, Sharmila Nangi, and Kuldeep Kulkarni. User-guided variable rate learned image compression. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2022. 2, 3

[23] Dailan He, Ziming Yang, Weikun Peng, Rui Ma, Hongwei Qin, and Yan Wang. 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 (CVPR)*, 2022. 1, 2, 3, 6

[24] Dailan He, Yaoyan Zheng, Baocheng Sun, Yan Wang, and Hongwei Qin. Checkerboard context model for efficient learned image compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2

[25] 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. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017. 6

[26] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020. 2

In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 1, 2

[27] Emiel Hoogeboom, Eirikur Agustsson, Fabian Mentzer, Luca Versari, George Toderici, and Lucas Theis. High-fidelity image compression with score-based generative models. arXiv preprint arXiv:2305.18231, 2023. 2, 6, 7

[28] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 3

[29] Alexia Jolicoeur-Martineau. The relativistic discriminator: a key element missing from standard GAN. In *International Conference on Learning Representations (ICLR)*, 2019. 4, 8

[30] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *International Conference on Learning Representations, (ICLR)*, 2014. 6

[31] Ahmet Burakhan Koyuncu, Han Gao, Atanas Boev, Georgii Gaikov, Elena Alshina, and Eckehard G. Steinbach. Contextformer: A transformer with spatio-channel attention for context modeling in learned image compression. In *European Conference on Computer Vision (ECCV)*, 2022. 2

[32] Alina Kuznetsova, Mohamad Hassan Mohamad 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. *International Journal of Computer Vision*, 128:1956–1981, 2020. 5

[33] Jooyoung Lee, Seunghyun Cho, and Seung-Kwon Beack. Context-adaptive entropy model for end-to-end optimized image compression. In *International Conference on Learning Representations (ICLR)*, 2019. 2

[34] Jinming Liu, Heming Sun, and Jiro Katto. Learned image compression with mixed transformer-cnn architectures. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 1, 2

[35] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10012–10022, October 2021. 2

[36] Yi Ma, Yongqi Zhai, Chunhui Yang, Jiayu Yang, Ruofan Wang, Jing Zhou, Kai Li, Ying Chen, and Ronggang Wang. Variable rate roi image compression optimized for visual quality. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2021. 2, 3

[37] Fabian Mentzer, Eirikur Agustsson, and Michael Tschannen. M2t: Masking transformers twice for faster decoding. arXiv preprint arXiv:2304.07313, 2023. 2

[38] Fabian Mentzer, George D Toderici, Michael Tschannen, and Eirikur Agustsson. High-fidelity generative image compression. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020. 1, 2, 3, 5, 6, 7

[39] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *European Conference on Computer Vision (ECCV)*, 2020. 3

[40] David Minnen, Johannes Ballé, and George Toderici. Joint autoregressive and hierarchical priors for learned image compression. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018. 2

[41] David Minnen and Saurabh Singh. Channel-wise autoregressive entropy models for learned image compression. In *IEEE International Conference on Image Processing (ICIP)*, 2020. 2, 3, 6, 7, 8

[42] Yichen Qian, Xiuyu Sun, Ming Lin, Zhiyu Tan, and Rong Jin. Entroformer: A transformer-based entropy model for learned image compression. In *International Conference on Learning Representations (ICLR)*, 2021. 2

[43] Oren Rippel and Lubomir Bourdev. Real-time adaptive image compression. In *Proceedings of International Conference on Machine Learning*, 2017. 2

[44] Myungseo Song, Jinyoung Choi, and Bohyung Han. Variable-rate deep image compression through spatially-adaptive feature transform. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 1, 2, 8

[45] Zhenhong Sun, Zhiyu Tan, Xiuyu Sun, Fangyi Zhang, Yichen Qian, Dongyang Li, and Hao Li. Interpolation variable rate image compression. In *Proceedings of ACM International Conference on Multimedia (ACMMM)*, 2021. 1, 2, 3, 6, 7

[46] Lucas Theis, Tim Salimans, Matthew D. Hoffman, and Fabian Mentzer. Lossy compression with gaussian diffusion. arXiv preprint arXiv:2206.08889, 2022. 1

[47] George Toderici, Lucas Theis, Nick Johnston, Eirikur Agustsson, Fabian Mentzer, Johannes Balle, Wenzhe Shi, and Radu Timofte. CLIC 2020: Challenge on learned image compression, 2020. https://www.tensorflow.org/datasets/catalog/clic. 6

[48] Fei Yang, Luis Herranz, Joost van de Weijer, José A. Iglesias Guitián, Antonio M. López, and Mikhail G. Mozerov. Variable rate deep image compression with modulated autoencoder. *IEEE Signal Processing Letters*, 27:331–335, 2020. 2

[49] Ruihan Yang and Stephan Mandt. Lossy image compression with conditional diffusion models. arXiv preprint arXiv:2209.06950, 2023. 2

[50] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 5, 6

[51] Yinhao Zhu, Yang Yang, and Taco Cohen. Transformer-based transform coding. In *International Conference on Learning Representations (ICLR)*, 2022. 2

[52] Renjie Zou, Chunfeng Song, and Zhaoxiang Zhang. The devil is in the details: Window-based attention for image compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2