# I2C: Invertible Continuous Codec for High-Fidelity Variable-Rate Image Compression

Shilv Cai, Liqun Chen, Zhijun Zhang, Xiangyun Zhao, Jiahuan Zhou, Yuxin Peng, *Senior Member, IEEE*, Luxin Yan, *Member, IEEE*, Sheng Zhong, and Xu Zou, *Member, IEEE*

***Abstract*—Lossy image compression is a fundamental technology in media transmission and storage. Variable-rate approaches have recently gained much attention to avoid the usage of a set of different models for compressing images at different rates. During the media sharing, multiple re-encodings with different rates would be inevitably executed. However, existing Variational Autoencoder (VAE)-based approaches would be readily corrupted in such circumstances, resulting in the occurrence of strong artifacts and the destruction of image fidelity. Based on the theoretical findings of preserving image fidelity via invertible transformation, we aim to tackle the issue of high-fidelity fine variable-rate image compression and thus propose the Invertible Continuous Codec (I2C). We implement the I2C in a mathematical invertible manner with the core Invertible Activation Transformation (IAT) module. I2C is constructed upon a single-rate Invertible Neural Network (INN) based model and the quality level (QLevel) would be fed into the IAT to generate scaling and bias tensors. Extensive experiments demonstrate that the proposed I2C method outperforms state-of-the-art variable-rate image compression methods by a large margin, especially after multiple continuous re-encodings with different rates, while having the ability to obtain a very fine variable-rate control without any performance compromise.**

***Index Terms*—Image coding, image processing, rate-distortion.**

## I. Introduction

LOSSY image compression is one essential problem especially in such an information explosion era, due to the increasing volume of visual data. A desired image compression method would effectively lower the data redundancy with fewer bits consumption while better preserving the image fidelity, for promoting applications of media sharing, storage, and processing. In order to achieve this goal, numerous classical image compression standards such as JPEG [2], JPEG2000 [3], Webp [4], BPG [5], AVIF [6], and Versatile Video Coding (VVC) [7] have been developed and widely utilized in various practical applications. Over recent years, remarkable progress has been made in learning-based image compression methods, which have shown superior performance in common metrics including PSNR and MS-SSIM. By exploiting the potent nonlinear transformation capabilities of DNNs, these methods [8], [9], [10] achieve end-to-end learning through a vast number of high-quality images, while minimizing the rate-distortion cost. Despite considerable advancements, learning-based image compression still poses challenges when it comes to adapting to variable-rate compression. Most existing approaches involve training multiple single-rate models for different rates, which results in high storage and training costs.

To remedy the issue, enabling variable-rate control within a single model based on the Variational Autoencoder (VAE) framework has attracted research interest [1], [11], [12], [13], [14], [15]. The researchers first try to achieve discrete rate adaptation using one single model. Choi et al. [11] introduced conditional convolution and achieved variable rate through two-stage training. Yang et al. [12] proposed the modulated autoencoder and achieved discrete adjustable compression rates by different Lagrange multipliers. Chen et al. [13] inserted a set of scaling factors directly before the quantizer to achieve the discrete adjustable compression rates. However, the performance of these methods would be dropped when conducting finer variable-rate control. Thus, the topic of fine rate adaptation has attracted more attention recently. Sun et al. [14] obtained continuously adjustable compression rate by linear interpolation. Cui et al. [15] achieved continuous compression rate control by exponential interpolation. Lin et al. [16] raised the scaling network, which is purposefully developed to convert the scalar value of the Lagrange multiplier to a vector, in order to scale the feature map channel-wise and achieve variable rate adaptation. Song et al. [1] conditioned on the quality map and achieved the variable rate, which requires semantic segmentation labels for training. Though these methods have the ability of fine variable-rate compression control, they need additional gain modules or semantic labels to maintain the performance.

Besides, in the current social-networking epoch, images would be shared and transmitted multiple times among

Manuscript received 10 April 2023; revised 12 December 2023; accepted 13 January 2024. Date of publication 22 January 2024; date of current version 7 May 2024. This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grants 62176100, 62301228, 61925201, 62132001, and 62376011, and in part by the Special Project of Science and Technology Development of Central Guiding Local of Hubei Province under Grant 2021BEE056. Recommended for acceptance by V. Morariu. *(Corresponding author: Xu Zou.)*

Shilv Cai, Liqun Chen, Zhijun Zhang, Luxin Yan, Sheng Zhong, and Xu Zou are with the National Key Laboratory of Multispectral Information Intelligent Processing Technology, School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan, Hubei 430074, China (e-mail: caishilv@hust.edu.cn; chenliqun@hust.edu.cn; zhangzhijun@hust.edu.cn; yanluxin@hust.edu.cn; zhongsheng@hust.edu.cn; zx@zoux.me).

Xiangyun Zhao is with Waymo, Mountain View, CA 94043 USA (e-mail: zhaoxiangyun915@gmail.com).

Jiahuan Zhou and Yuxin Peng are with the Wangxuan Institute of Computer Technology, Peking University, Beijing 100871, China (e-mail: jiahuanzhou@pku.edu.cn; pengyuxin@pku.edu.cn).

The project is publicly available at https://github.com/CaiShilv/HiFi-VRIC.

Digital Object Identifier 10.1109/TPAMI.2024.3356557

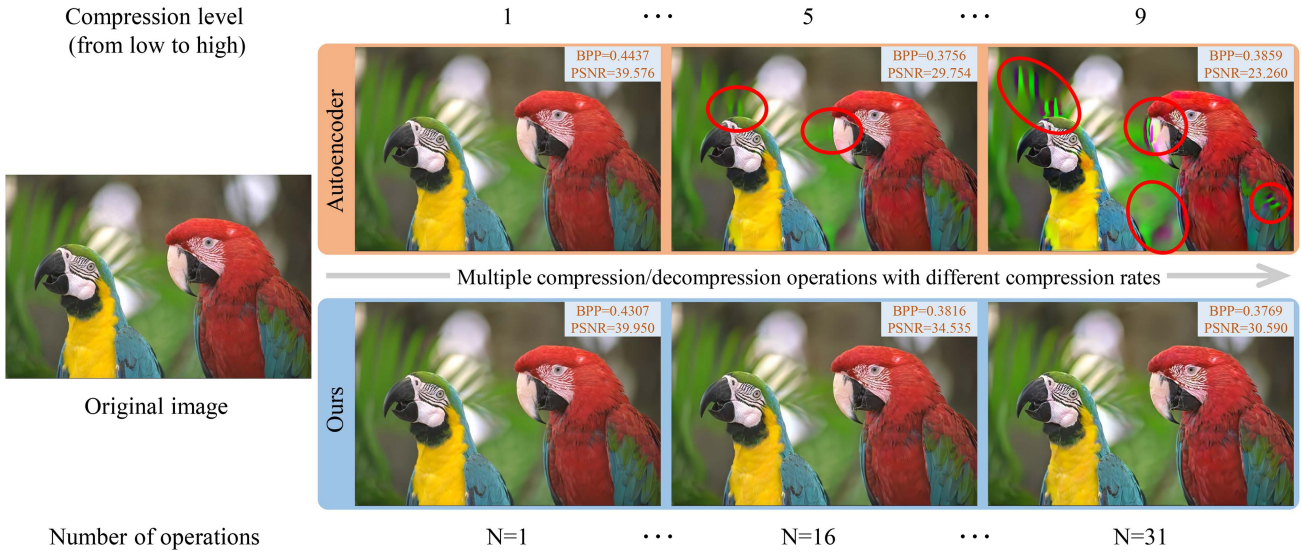



Fig. 1. Reconstructed images of variable-rate compression methods after different numbers of compression/decompression operations. It broadly occurs in image sharing among social platforms. Severe artifacts and color shifts would appear (see regions in red circles) in the state-of-the-art VAE-based approach [1] once multiple continuous re-encodings are executed, in contrast to fewer artifacts and higher fidelity results achieved by our proposed approach. High-fidelity preserving with fine variable-rate control is the main advantage and novelty of our work.

numerous entities (e.g., one person may download a compressed image from Facebook and then send it to his friend via WeChat under another re-encoding). It would be particularly interesting for a variable-rate image codec if the fidelity of images could be preserved once continuous compression/decompression operations are executed. However, state-of-the-art VAE-based variable-rate approaches (e.g., Song et al. [1]) would be readily corrupted once multiple continuous re-encodings are executed, resulting in the fact that image quality would be tremendously dropped. Strong artifacts and color shifts would appear, as shown in Fig. 1. The main reason is that the autoencoder transforms the image to a low-dimensional latent space and irreversibly discards information before quantization, imposing an implicit limitation on the reconstruction quality. To alleviate information loss, Invertible Neural Networks [17], [18] have gained much attention to effectively preserve fidelity. It is worth noting that VAE-based variable-rate methods cannot be directly fused into the INN-based framework since implementations of their variable-rate control do not satisfy the bijective mapping property. Whether it is possible to introduce conditional control in the INN-based framework to achieve variable rate has not been analyzed and derived from a theoretical basis so far.

In this paper, we first carry out in-depth theoretical analyses and mathematical derivations of condition-based invertibility. Based on the exploration of conditional control invertibility, we propose the Invertible Continuous Codec (I2C). Invertible Activation Transformation (IAT) is the core module of I2C that exhibits a mathematical invertible property to avoid discarding any information in the latent space to preserve high fidelity. I2C is constructed upon a single-rate Invertible Neural Network (INN) based model and the quality level (QLevel) would be fed into the IAT to generate scaling and bias tensors. IAT and QLevel together give I2C the ability of fine variable-rate control while better preserving the image fidelity. We initially extend the mathematical invertibility to the variable-rate image compression. Moreover, the proposed image compression method attempts to achieve finer control of multiple variable rates, by presenting a compatible tensor-based Lagrange multiplier to train the whole model. The contributions of our proposed method are 4-folded:

- We propose an effective yet neat variable-rate image compression method named Invertible Continuous Codec (I2C) under the design of a conditional invertible manner to achieve the high fidelity of reconstructed images, especially after multiple continuous variable-rate image compression/decompression operations. This issue is rarely investigated so far.
- In-depth theoretical analyses and mathematical derivations of condition-based invertibility are provided. With the theoretical foundation to support it, the proposed I2C can be easily applied to different INN frameworks.
- I2C achieves fine variable-rate control without any performance compromise by storing only byte-level additional information in the bitstream directly (e.g., once 2 bytes are adopted, $2^{16} = 65536$ effective fine variable rates are achieved).
- Extensive experiments demonstrate the superiority of our proposed I2C in image fidelity preserving, rate-distortion performance, and fine rate adaptation over three datasets, including Kodak [19], CLIC [20], and DIV2K [21]. Besides, we conduct comparison experiments on practical biomedical and remote sensing images to show the application potential of it.

This manuscript is an extension of our previous conference paper [22], while we have made plenty of extensions including 1) Thorough theoretical analyses and mathematical derivations of the condition-based invertibility, which is the footing stone of our high-fidelity fine variable-rate image continuous

re-encodings, are provided. With the theoretical foundation to support it, our proposed I2C can be easily applied to different INN frameworks. We present the rationality of I2C, which can be considered as a kind of conditional invertible neural network, for variable-rate image compression. We also present why the IAT module can be integrated with affine coupling layers, to jointly construct I2C and implement the invertible design, finally forming conditional affine coupling layers. 2) The proposed I2C is additionally implemented into other three INN-based single-rate image compression architectures to evaluate the robustness of our proposed method. We use three different Invertible Neural Networks from Incompressible-flow Network (GIN) [23], NICE [24], and GLOW [25] instead of the original architecture, initially proposed in RealNVP [26] and adopted by the invertible block of baseline model [18], to verify that the proposed I2C can adapt to different INN-based architectures. I2C with different INN-based architectures are consistently able to achieve fine variable-rate control while preserving high fidelity. Such additional experiments show that I2C is a plug-and-play method that can be readily integrated into INN-based single-rate image compression frameworks to enhance their abilities without harming invertibility. 3) To further facilitate the codec efficiency and alleviate the computation burden, an optimized model is redesigned according to the complementary analyses of the characteristics of the network structure. Compared with the previous model, the number of parameters of this optimized version has been reduced by 1/3 without any performance compromise. 4) Supplementary experiments on finer variable-rate control are conducted. The mechanism of achieving such ability is analyzed and discussed in depth. The analysis shows that the small amount of additional bitstream storage can generate a large number of fine variable rates, which is quite practical in real-world applications. 5) Variable-rate multiple continuous re-encodings comparison experiments with 6 more single-rate typical image compression methods are further carried out to demonstrate the superiority of our proposed method. We also add the comparison experiment with the variable rate method of Lin et al. [16] to validate the effectiveness of our proposed method further. Besides, we conduct supplementary comparison experiments on practical biomedical and remote sensing images to show its application potential.

## II. Related Work

In recent years, the application of neural networks in image compression has attracted widespread attention. The Variational Autoencoder (VAE) [8], [9], [10], [27], [28], [29], [30], [31], [32], [33], [34], [35], [36], [37], [38], [39], [40], [41], [42], [43], Invertible Neural Network (INN) [17], [18], [44], [45] and Generative Adversarial Networks (GAN) [46], [47], [48], [49], [50] based methods have achieved surprising results.

### A. Learned Single-Rate Image Compression

The VAE-based framework is used as a nonlinear transformation coding model, which is the main approach in the learned image compression method. The works [8], [9], [10] were the first to use CNN for end-to-end image compression and inspired many learning-based image compression methods. The work [10] introduced a hyperprior entropy model to capture the zero-mean Gaussian distribution of the latent representations. The works [27], [29] used the Gaussian model with the non-zero mean to improve the ability to model latent representations. Later works [27], [28], [29] further removed redundancy in potential features using the context model. By using the global similarity within the context, Li et al. [39] proposed a special non-local operation for context modeling. To exploit a serial decoding process for causal contextual entropy prediction in latent space, Guo et al. [40] proposed the concept of separate entropy coding. Further, the 3D-context entropy model [30], multi-scale hyperprior entropy model [32], and discretized Gaussian mixture model [33] were used to further improve the entropy model. In addition, channel-wise module [34], attention module [33], [35], non-local attention module [36], [51], and content-weighted methods [43], [52], [53] were used to extract better latent representations. Recently transformer was used to capture long-range dependencies in probability distribution estimation effectively and efficiently [54], [55], [56]. The first successful attempt to apply a transformer-based method to image compression was made by Qian et al. [54]. Lu et al. [56] proposed the neural transformation unit as the basic module. It consists of a Swin Transformer block and a convolutional layer for better information embedding. In addition, Cai et al. [43] proposed the deep learning-based unified framework that allows for rate-distortion optimization for ROI image compression. Abhijith Punnappurath and Michael S. Brown investigated the ability of deep image compressors to be ”aware” of the additional goal of raw reconstruction. David et al. [42] proposed a two-step learning-based image compression method to build convolution neural networks for the analysis of gigapixel images using only weak labels at the image level.

Most learning-based image compression methods need to train different network models for various compression rates, which not only increases the storage of computational resources but also is not compatible with practical applications. Therefore, using one single model to achieve variable rate adaptation was widely studied.

### B. Learned Variable Rate Image Compression

Initially, LSTM networks [57], [58], [59] control different compression rates by the different number of iterations. The more iterations, the clearer the reconstructed image would be. However, the LSTM-based approach cannot outperform JPEG2000 [3] in rate-distortion performance and would not obtain continuous compression rates. LSTM-based approaches use a large number of $3 \times 3 \times 512 \times 512$ network layers, such a structure can make the network computationally slow. In addition, the iterative procedure is also time-consuming. Thus it is not suitable for practical applications. Then Choi et al. [11] introduced conditional convolution in the autoencoder framework to achieve variable-rate adaptation with a single model through two-stage training.

However, while the variable rate is achieved, the rate-distortion performance degrades and there is a dilemma in

choosing the appropriate Lagrange multiplier and quantization step size for forward inference. Yang et al. [12] proposed a modulated autoencoder that achieved the discrete adjustable compression rate with a single model by different Lagrange multipliers. Thesis et al. [9] first trained the model with high bits per pixel (bpp) and then fixed the network model parameters to train the scaling parameters for different compression rates. However, the network model suffered from incongruity with the scaling parameters, especially in low bpp cases. Chen et al. [13] inserted a set of scaling factors directly before the quantizer to achieve the discrete variable compression rate. Mei et al. [38] proposed an end-to-end optimized quality and spatial scalable image compression model (QSSIC) to achieve variable rates.

Recently, research has been conducted on continuous compression rate adjustable [1], [14], [15], [16]. The work [15] introduced a series of vector pairs for coarse compression rate control and then achieved continuous compression rate control by exponential interpolation. Lin et al. [16] used the scaling network, which is designed to map the scalar value of the Lagrange multiplier into a vector, to scale the feature map channel-wise achieving the variable compression rate. Sun et al. [14] extended the work [11], which obtained a continuously adjustable compression rate by linear interpolation. Song et al. [1] conditioned the quality map by spatial feature transform (SFT) [60] to control different compression rates.

VAE-based variable-rate approaches have been extensively researched. However, those methods suffer from severe information distortion after multiple continuous operations of compression/decompression for the same image. The distortion becomes more explicit as the number of operations increases.

### C. Invertible Neural Networks

Invertible neural networks (INNs) are generative models that transform complex distributions into simple ones, allowing for accurate and efficient probability density estimation. INNs have a bijective mapping of input and output, which is ideal for image compression.

NICE [24] introduced a flexible architecture that can learn highly nonlinear bijective transformations to represent data with simple distributions. Based on NICE [24], RealNVP [26] further extended the idea of hierarchical and combinatorial transformations, which used affine coupling and a multi-scale framework. Kingma et al. [25] proposed a generative flow model based on a $1 \times 1$ invertible convolutional network with a significant improvement in log-likelihood on a standard benchmark dataset, having the advantages of exact controllability of log-likelihood, the tractability of exact inference of latent representations, and parallelizability of training and synthesis. It shows that a generative model optimized for the simple log-likelihood objective is capable of efficiently synthesizing and manipulating large images in a realistic way. Ardizzone et al. [61] demonstrated that the validity of INNs is suitable not only for synthetic data but also for two practical applications in medicine and astrophysics. Sorrenson et al. [23] generalized the theory to the case of an unknown intrinsic problem dimension, proving that in some special (but not very restrictive) cases, informative latent variables are automatically separated from noise by an estimator. SRFlow [62] has designed a conditional normalizing flow architecture to solve the ill-posed problem in the super-resolution task. Xiao et al. [63] proposed an invertible rescaling network (IRN), which constructed a bijective transform to effectively implement the reconstruction of low-resolution into high-resolution images.

INN greatly alleviates the information loss problem for better image compression, as in [17], [18], [44], [45], [64]. But no one has specifically studied variable-rate image compression with a single model based on the INN framework.

## III. Methodology

### A. Framework

Our image compression approach I2C is depicted in Fig. 2. The proposed method implements fine variable-rate modulation in an invertible neural network framework, which involves the invertible activation transformation (IAT) module to control different compression rates through different quality levels. We present the detailed procedure of the model in the following: First, the source image $x \in \mathbb{R}^{3\times H\times W}$ is enhanced by the dense block module [65] to generate a nonlinear representation of $u \in \mathbb{R}^{3\times H\times W}$, where $H$ and $W$ denote the height and width of the input image respectively. Then the forward pass of the Invertible Neural Network section, which is equipped with the proposed IAT module, transforms $u$ to a latent representation, conditioned on the quality level $L \in \mathbb{R}^{H\times W}$ to control the compression rate. This latent representation would be further fed into the Attention Channel Squeeze module to reduce the number of channels and obtain the potential representation $y$. This procedure could be formulated by a parametric analysis transform function, *i.e.:*

$$y = g_a(x, L), \tag{1}$$

the discrete latent features $\hat{y}$ are obtained by quantification of $y$, i.e., $\hat{y} = Q(y)$. We use the quantizer $Q(\cdot)$ in Ballé et al. [10] to model the quantized latent representation $\hat{y}$ approximately by adding the uniform noise $U(-0.5, 0.5)$ to the latent representation $y$ during training and rounding the latent representation $y$ during testing. The context entropy model generates parameters $\mu$ and $\sigma$ of the Gaussian entropy model that approximates the distribution of quantified latent representation $\hat{y}$ to support the entropy encoding. We use range asymmetric numeral system [66] to losslessly compress latent representation $\hat{y}$ and $\hat{z}$ into bitstreams.

The inverse calculation takes the quantified latent representation $\hat{y}$ and the quality level $L$ as the input, and reconstructs the decompressed images by a parametric synthesis transform, which is formulated as follows:

$$\hat{x} = g_s(\hat{y}, L). \tag{2}$$

### B. Invertible Activation Transformation

We proposed the invertible activation transformation (IAT) module to enhance the invertible neural network, which efficiently generates the desired compressed representation conditional on the quality level $L$. The proposed IAT module can

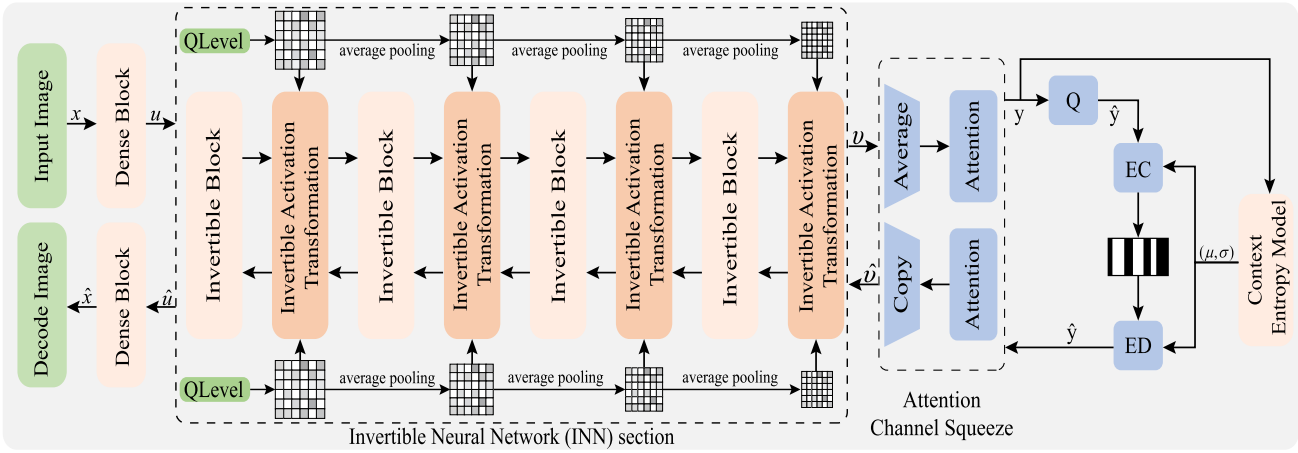



Fig. 2. Framework of I2C. We insert IAT into the Invertible Neural Network section and utilize it to generate element-wise activation parameters of features from the input quality level (QLevel). IAT and QLevel together give I2C the ability of fine variable-rate control while preserving the image fidelity especially when multiple continuous compression/decompression operations are executed. EC/ED means entropy encoding/decoding respectively. Q is the quantizer. Parameters ($\mu$, $\sigma$) of the context entropy model are used to support EC/ED.

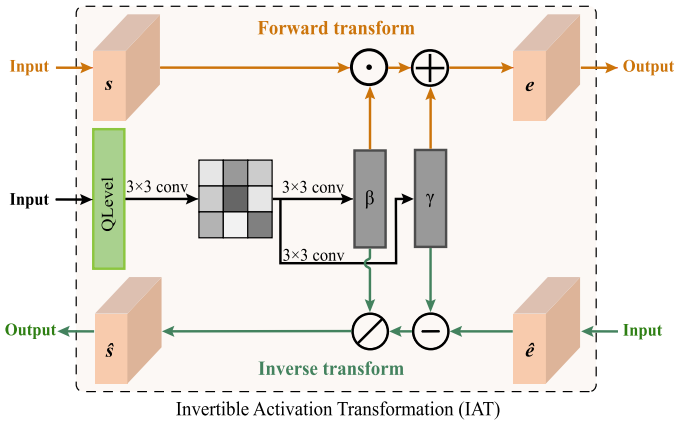



Fig. 3. Illustration of the IAT module. The forward and inverse transformation of the IAT module implements the bijective mapping. This module takes the QLevel and feature as input to generate element-wise activation parameters $\beta$ and $\gamma$, further obtaining the output results. Thus, the forward and inverse procedures are mathematically invertible, enhancing the fidelity of reconstructed images.

achieve variable-rate adaption on a single model while preserving the image fidelity, especially after multiple continuous compression/decompression operations, in a mathematical invertible manner.

The forward transform of the IAT module is illustrated by pink arrows on the top of Fig. 3. The inputs are the quality level $L$ and the feature $\boldsymbol{s}$. The element-wise activation parameters $\gamma \in \mathbb{R}^{c\times h\times w}$ and $\beta \in \mathbb{R}^{c\times h\times w}$ are then calculated by the IAT module from the quality level $L$ via convolutional operations. These activation parameters would be applied to the feature $\boldsymbol{s}$ via the (3) to generate the feature $\boldsymbol{e}$:

$$\boldsymbol{e} = (\boldsymbol{s} \odot \beta) \oplus \gamma, \tag{3}$$

where $\odot$ denotes the Hadamard product and $\oplus$ denotes the addition by element. $c$, $h$, and $w$ are the channel, height, and width of the feature, respectively.

The inverse transform of the IAT module is illustrated by green arrows at the bottom of Fig. 3. The input quality level $L$ and features $\hat{\boldsymbol{e}}$ are applied to obtain the feature $\hat{\boldsymbol{s}}$. This inverse transform is formulated by (4):

$$\hat{\boldsymbol{s}} = (\hat{\boldsymbol{e}} \ominus \gamma) \oslash \beta, \tag{4}$$

where $\ominus$ denotes the subtraction in elemental order, $\oslash$ denotes the division by elemental order. Once the quality level $L$ is the same in both forward and inverse procedures, the invertibility of the operation between the features $\boldsymbol{s}$ and $\boldsymbol{e}$ can be guaranteed.

In the previous work [13], a set of scaling factors was inserted directly before the quantizer to achieve the discrete adjustable compression rate. In our algorithm, the activation parameters are element-wise, which means that the IAT module is computed as a spatial feature transform rather than a simple channel weighting. Moreover, the IAT module is attached after each invertible block which is initially proposed in RealNVP [26] and adopted by baseline model [18], not just inserted before the quantizer. These adjustments not only make fine variable-rate adaptation available but also turn out to better performance, the experiment "Impact of the QLevel Representation" in Section VI-B shows its effectiveness, and the results are shown in Fig. 15.

### C. Fine Variable-Rate Control

Unlike interpolation-based methods [14], [15] for obtaining finer compression rates, our method achieves the fine compression rate adaptation directly by modulating the quality level $L$, which is more convenient when controlling the compression rate by only one parameter instead of two. Compared to Song et al. [1], our method does not require additional semantic labels, either.

The goal of lossy image compression is to minimize the length of the bits stream and the distortion between the source image $x$ and the reconstructed image $\hat{x}$. The optimization function is always expressed in the rate-distortion:

$$L = R + \lambda D, \tag{5}$$

where λ is the Lagrange multiplier which determines the trade-off between the rate $R$ and the distortion $D$. Theoretically, as long as the set of Lagrangian multiplier λ is large enough, it is possible to achieve fine compression rate control, but in practice, the computational cost is too high. For interpolation-based methods, the Lagrangian multiplier λ is a scalar. Thus, at each iteration during training, only one element in a finite set of λ would be randomly selected for optimization. In order to further promote the R-D performance of our model, we use a tensor instead of the scalar λ. Our optimization function implements fine variable-rate control by minimizing the rate-distortion:

$$Loss = R + \Lambda \odot \mathbf{D}, \tag{6}$$

where dimensions of $\Lambda \in \mathbb{R}^{C\times H\times W}$ and the distortion $\mathbf{D} \in \mathbb{R}^{C\times H\times W}$ are the same as the dimension of the original input image. $\odot$ denotes the Hadamard product. In this formulation, $\Lambda$ is a tensor and no longer a finite set of constant scalars. Thus, $\mathbf{D}$ measures pixel-wise distortion and is defined as $\mathbf{D} = \frac{\sum_{i=1}^{T}\lambda_i(x_i-\hat{x}_i)^2}{T}$, $T$ indicates the number of image pixels, $\lambda_i$ is the Lagrangian multiplier, $x_i$ and $\hat{x}_i$ denote one pixel of the original image $x$ and reconstructed image $\hat{x}$, respectively.

$\Lambda$ is simply calculated from the quality level L via a monotonically increasing function: $\Lambda = V(L)$, where $V : \mathbb{R}^N \to \mathbb{R}^T$. $V(L) = \theta \times e^{\tau\times L}$, $\theta = 0.0012$, $\tau = 4.382$, the process of dimensioning from $\mathbb{R}^N \to \mathbb{R}^T$ is done by direct replication between channels. $L = [l_i]_{i=1:N}, l_i \in [0, 1], N = H \times W$, $T = C \times H \times W$. $C$, $H$, and $W$ denote the channel, height, and width of the source image $x$, respectively. Under such a paradigm, we implement this pixel-wise distortion constraint by randomly generating values of each element of the tensor $\Lambda$ via the quality level $L$ during training. This is equivalent to increasing the number of λ values selected at each iteration. So, the fine variable-rate control can be obtained by feeding exact quality levels during the testing.

As in other learning-based method [10], the log-likelihood of the coded features $\hat{y}$ is estimated by a probabilistic model to replace the true compression rate $R$. Finally, the training loss would be:

$$\begin{aligned} Loss = & -\log_2 P_{\hat{y}}(\hat{y}|x,\Lambda) - \log_2 P_{\hat{z}}(\hat{z}|x,\Lambda) \\ & + \frac{\sum_{i=1}^{T}\lambda_i(x_i-\hat{x}_i)^2}{T}, \end{aligned} \tag{7}$$

where $\hat{y}$ and $\hat{z}$ are quantized latent representations and side information respectively. $p_{\hat{y}}(\hat{y}|x,\Lambda) = \mathcal{N}(\mu,\sigma^2)$, $\mu$ and $\sigma$ denote the estimates of the mean and standard deviation of the quantified latent representation $\hat{y}$. $p_{\hat{z}}(\hat{z}|x,\Lambda) = \mathcal{N}(\mu_1,\sigma_1^2)$, $\mu_1$ and $\sigma_1$ denote the estimates of the mean and standard deviation of the quantified side information $\hat{z}$. The side information usually represents the hyperprior originally proposed in [10] and refers to the extra stream $\hat{z}$ generated by the "Context Entropy Model" in Fig. 2. It is worth noting that this loss function would be degraded to the standard rate-distortion optimization function if all elements of the tensor quality level $L$ are the same.

In addition, our method can be trained on arbitrary unlabeled data instead of requiring semantic segmentation labels corresponding to the original data, which is different from Song et al. [1], for training the model.

## IV. Theoretical Analyses and Derivations

Thanks to the invertible design, I2C can better preserve image fidelity, especially after multiple continuous re-encodings with different compression rates. Here, we present the mathematical derivation of such a design and show why fidelity preservation works.

### A. Conditional Invertible Neural Network for Lossy Image Compression

In this subsection, we would like to present the rationality of I2C, which can be considered a kind of conditional invertible neural network, for variable-rate image compression. Lossy image compression usually can be divided into three modularized components: transform, quantization, and entropy coding. The goal of lossy image compression is to transform the original image $x$ to symbols $\hat{y}$ to be entropy coded. Typical learned single-rate image compression approaches learn a deterministic mapping $x \mapsto \hat{y}$ when given the trade-off λ in (5). We aim to get the conditional distribution $P_{\hat{y}|x}(\hat{y}|x,\lambda)$, and different mappings from $x$ to $\hat{y}$ are achieved by different λ. Finally, different compression rates can be obtained by different λ.

The key idea of the invertible neural network (INN) [24], [26] is to parameterize the distribution $p_{v|u}$ by the INN $f_\phi$. When introducing conditional settings, $f_\phi$ makes the deterministic mapping to the variable latent representation $v = f_\phi(u,\lambda)$. If the function $f_\phi$ is invertible, the original feature $u$ can be obtained from the latent representation $v$ as $u = f_\phi^{-1}(v,\lambda)$. The core aspect of the invertible neural network is that the probability density $p_{v|u}$ can be explicitly computed as:

$$p_{v|u}(v|u,\lambda,\phi) = p_{u|v}(f_\phi^{-1}(v,\lambda))\left|\det\frac{\partial f_\phi^{-1}(v,\lambda)}{\partial v}\right|^{-1}. \tag{8}$$

It is derived by applying the change-of-variables formula for densities, where the second factor is the resulting volume scaling given by the determinant of the Jacobian $\frac{\partial f_\phi^{-1}(v,\lambda)}{\partial v}$. The (8) allows us to train the network by optimizing $\mathfrak{L}$ through minimizing the negative log-likelihood for training the invertible function $f_\phi$:

$$\begin{aligned} \mathfrak{L}(\phi;u,v,\lambda) &= -\log p_{v|u}(v|u,\lambda,\phi) \\ &= -\log p_{u|v}(f_\phi^{-1}(v,\lambda)) - \log\left|\det\frac{\partial f_\phi^{-1}(v,\lambda)}{\partial v}\right|^{-1}. \end{aligned} \tag{9}$$

In this formulation, for preventing the collapse of the latent space, the Jacobian log-determinant is adopted inspired by [45]. In our implementation, we can use the reconstruction item $d(u,\hat{u})$ instead of it. In the Fig. 2, when we consider combining dense block and attention channel squeeze, $d(x,\hat{x})$ is involved in the distortion item. Meanwhile, the side information $\hat{z}$ of the

entropy model should be considered. That is, the total loss could be formulated as:

$$Loss = -\log p_{\hat{y}}(\hat{y}|x,\lambda) - \log p_{\hat{z}}(\hat{z}|x,\lambda) + \lambda d(x,\hat{x}). \quad (10)$$

### B. Conditional Affine Coupling Layers

In this subsection, we present why the IAT module can be integrated with affine coupling layers [24], [26], to jointly construct the I2C and implement the invertible design. The invertible neural network $f_\phi$ can be decomposed into a sequence of invertible layers. In fact, combinations of the IAT module and the affine coupling layer, which is contained in the invertible block in Fig. 2, can compose a sequence of conditional affine coupling layers. The $i$th conditional affine coupling layer takes an input $u_{1:C}^{(i)}$ with dimensional size of $C$. It splits the inputs at $c$th channel into two parts and gets the output $u_{1:C}^{(i+1)}$ with the channel dimension of $C$ under the condition $\lambda$:

$$\Theta_{1:c}^{(i+1)} = u_{1:c}^{(i)} \odot \exp(\sigma_c(g_2(u_{c+1:C}^{(i)}))) + h_2(u_{c+1:C}^{(i)}), \quad (11)$$

$$u_{1:c}^{(i+1)} = \Theta_{1:c}^{(i)} \odot \beta_{1:c}^{(i+1)} + \gamma_{1:c}^{(i+1)}, \quad (12)$$

$$\Psi_{c+1:C}^{(i+1)} = u_{c+1:C}^{(i)} \odot \exp(\sigma_c(g_1(u_{1:c}^{(i+1)}))) + h_1(u_{1:c}^{(i+1)}), \quad (13)$$

$$u_{c+1:C}^{(i+1)} = \Psi_{c+1:C}^{(i+1)} \odot \beta_{c+1:C}^{(i+1)} + \gamma_{c+1:C}^{(i+1)}, \quad (14)$$

where $\odot$ denotes the Hadamard product, $\exp(\cdot)$ denotes the exponential function, and $\sigma_c(\cdot)$ denotes the sigmoid function. The $\beta^{(i)}$ and $\gamma^{(i)}$ are calculated by the condition $\lambda$ (details are mentioned in Section III-B). $g_1$, $g_2$, $h_1$, and $h_2$ can be any feedforward functions and need not be invertible. During the inverse processing, the $i$th conditional affine coupling layer inversely takes $u_{1:C}^{(i+1)}$ as input and split it at $c$th channel. The conditional affine coupling layer gives a perfect inverse:

$$\Psi_{c+1:C}^{(i)} = (u_{c+1:C}^{i+1} - h_1(u_{1:c}^{(i+1)})) \odot \exp(-\sigma_c(g_1(u_{1:c}^{(i+1)}))), \quad (15)$$

$$u_{c+1:C}^{(i)} = (\Psi_{c+1:C}^{(i)} - \gamma_{c+1:C}^{(i)}) \oslash \beta_{c+1:C}^{(i)}, \quad (16)$$

$$\Theta_{1:c}^{(i)} = (u_{1:c}^{(i+1)} - h_2(u_{c+1:C}^{(i)})) \odot \exp(-\sigma_c(g_2(u_{c+1:C}^{(i)})))), \quad (17)$$

$$u_{1:c}^{(i)} = (\Theta_{1:c}^{(i)} - \gamma_{1:c}^{(i)}) \oslash \beta_{1:c}^{(i)}, \quad (18)$$

where $\oslash$ denotes the division by elemental order. Through the above equations, the invertibility is inherently guaranteed by the mathematical design. When the features are calculated in these conditional affine coupling layers, the information will not be lost, and the fidelity of the information will be elegantly preserved.

## V. Experiments

### A. Implementation Details

*Details For Training:* In our implementation, the network of Xie et al. [18] is adopted as our basic architecture. The training datasets contain Flicker 2W [67] and COCO [68]. Our network is trained on $256 \times 256$ randomly cropped patches and discards images less than 256px in height or width during data pre-processing. All experiments are conducted in the RGB space. In training, the quality level $L$ needs to be sent to the INN section as a condition during the forward and inverse transform. The quality level $L$ takes a uniform value tensor between (0,1) during the testing and is randomly sampled between (0,1) during the training. Our implementation relies on Pytorch [69] and an open-source CompressAI PyTorch library [70]. All experiments were conducted on RTX 3090 GPU and trained for about 2.5 M iterations with batch size 8. Adam optimizer [71] is used to optimize the parameters, there were multistage learning rates $\{1e-4, 5e-5, 1e-5, 5e-6, 1e-6, 5e-7\}$ that changed with boundaries $\{1000000, 1300000, 1600000, 1900000, 2200000, 2500000\}$.

*Details For Testing:* We evaluate the rate-distortion performance on three commonly used datasets. The Kodak [19] contains 24 lossless images with a size of $768 \times 512$. The CLIC Professional Validation dataset [20] comprises 41 high-quality images with much higher resolution. The DIV2K validation dataset [21] contains 100 images with high resolutions of 2 K. We draw curves based on the rate-distortion performance to compare the coding efficiency of different methods. We also calculate the area under the rate-distortion curve to observe the performance difference more effectively.

### B. Fidelity for Re-Encoding

In order to verify the ability of high fidelity preserving of I2C, our method is compared with the latest VAE-based variable-rate method proposed by Song et al. [1] according to their official codes. Since their method does not use a context model, we remove the context model and add the non-local attention module [36] in the hyperprior layer for our approach to make a fair comparison.

Fig. 4(a) and (c) show the performance after multiple continuous operations of compression/decompression with different compression rates. Both approaches change from high to low bpp ranges. Our I2C adopts bpp in the set of {1.027, 1.027, 1.012, 1.012, 0.995, 0.995, 0.978, 0.978, 0.962, 0.962, 0.946, 0.946, 0.929, 0.929, 0.913, 0.913, 0.897, 0.897, 0.881, 0.881, 0.866, 0.866, 0.851, 0.851, 0.836, 0.836, 0.821, 0.821, 0.806, 0.806, 0.791, 0.791} and Song et al. [1] adopts bpp in the set of {1.039, 1.039, 1.025, 1.025, 1.009, 1.009, 0.993, 0.993, 0.977, 0.977, 0.961, 0.961, 0.945, 0.945, 0.929, 0.929, 0.913, 0.913, 0.897, 0.897, 0.881, 0.881, 0.866, 0.866, 0.851, 0.851, 0.835, 0.835, 0.820, 0.820, 0.805, 0.805}. It is clearly seen that our method outperforms Song et al. [1] by a large margin, after multiple continuous variable-rate re-encodings. Fig. 4(b) and (d) show the performance by multiple operations with the fixed compression rate. Both approaches achieve a bit rate of 0.791 bpp for all steps. Also, our method achieves better results significantly, compared with Song et al. [1] and baseline [18]. In Fig. 4 (b) and (d), our model outperforms the fixed-rate baseline [18] and the variable-rate Song et al. [1] by a large margin on both PSNR and MS-SSIM. The results indicate that our proposed IAT module is powerful to maintain image fidelity, which is important for practical applications.

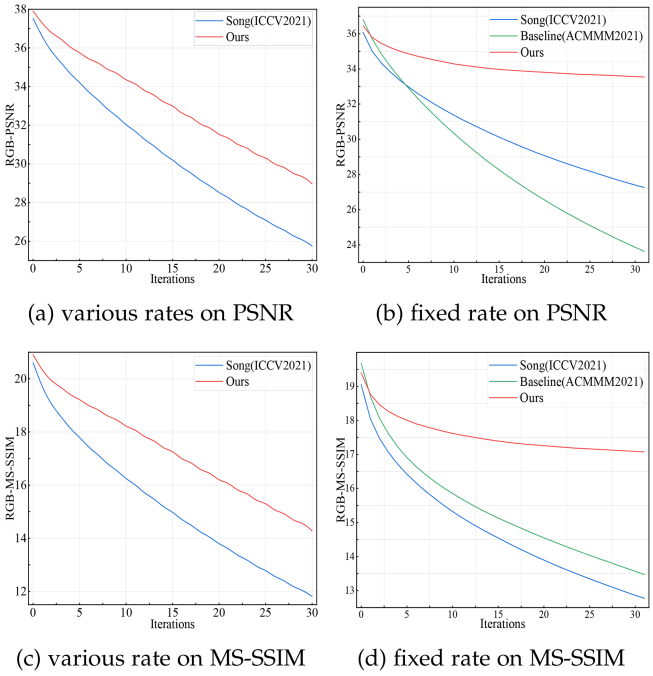



Fig. 4. Successive re-encodings on the Kodak dataset. (a) and (c): Compression rates of each compression/decompression operation are different. (b) and (d): The compression rate is fixed. Our approach outperforms baseline [18] and Song et al. [1] (a SOTA variable-rate approach) by a large margin to show the superiority of fidelity preserving especially when multiple continuous operations are executed.

Fig. 5(a) and (b) show the performance between our I2C and Ballé et al. [10] after multiple operations of compression/decompression with different compression rates. Both approaches change from high to low bpp ranges with the bpp set of {0.939, 0.669, 0.478, 0.320, 0.209, 0.131}. Fig. 5(c) and (d) show the performance between our I2C and Minnen et al. [29] with the bpp set of {0.885, 0.639, 0.432, 0.288, 0.187, 0.111}. Fig. 5(e) and (f) show the performance between our I2C and Chen et al. [36] with the bpp set of {0.859, 0.623, 0.419, 0.274, 0.177}. Fig. 5(g) and (h) show the performance between our I2C and Hu et al. [32] after multiple re-encodings with different compression rates. Both approaches change from high to low bpp ranges with a bpp set of {0.796, 0.411, 0.309, 0.208}. Fig. 6(a) and (b) show the performance comparison between our proposed I2C and Qian et al. [54] with bpp set of {0.931, 0.593, 0.406, 0.263, 0.145}. Fig. 6(c) and (d) show the performance comparison between our proposed I2C and Lu et al. [56] with bpp set of {0.864, 0.614, 0.431, 0.286, 0.185, 0.112}. It is clearly seen that our proposed I2C consistently outperforms those learning-based (including VAE and Transformer) methods by a large margin, after multiple continuous variable-rate re-encodings. We also conduct extra apple-to-apple experiments on the CLIC dataset. The experimental results are shown in Table I. The results also indicate that our proposed IAT module is powerful to maintain image fidelity, which is important for practical applications.

Fig. 7 illustrates the results of multiple continuous re-encodings on the same image with different compression rates.

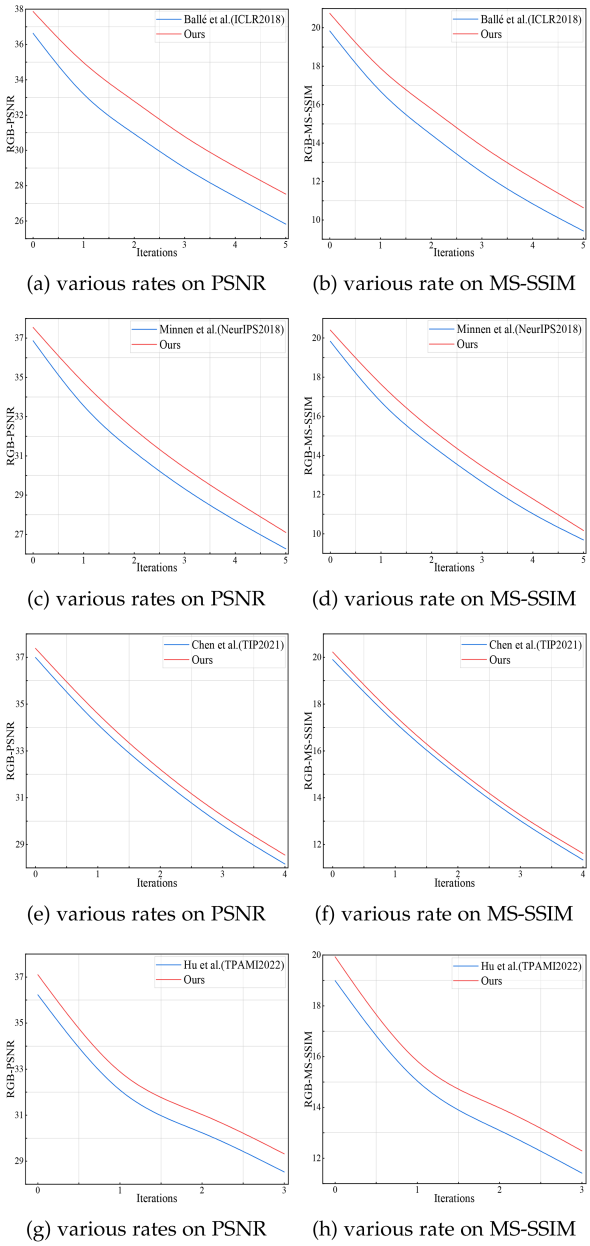



Fig. 5. Multiple continuous re-encodings on the Kodak dataset [19]. The compression rates of each compression/decompression operation are changed. Our approach compares with several typical single-rate image compression methods including (a) and (b) Ballé et al. [10], (c) and (d) Minnen et al. [29], (e) and (f) Chen et al. [36], (g) and (h) Hu et al. [32] on metrics of PSNR and MS-SSIM respectively. Our approach outperforms them by a large margin to show the superiority of fidelity preserving especially when multiple variable-rate re-encodings are executed. Besides, it is worth noting that those single-rate models need different models to adapt to various compression rates while our I2C is a variable-rate method.

With operations increasing, our proposed method shows higher fidelity while the VAE-based method [1] gradually raises severe artifacts and color shifts. Fig. 8 is the visualization results showing those single-rate image compression methods compared with our I2C during the variable-rate re-encodings on same images. All images of the single-rate methods are the results of the corresponding last iteration in Fig. 5. It is noted that the

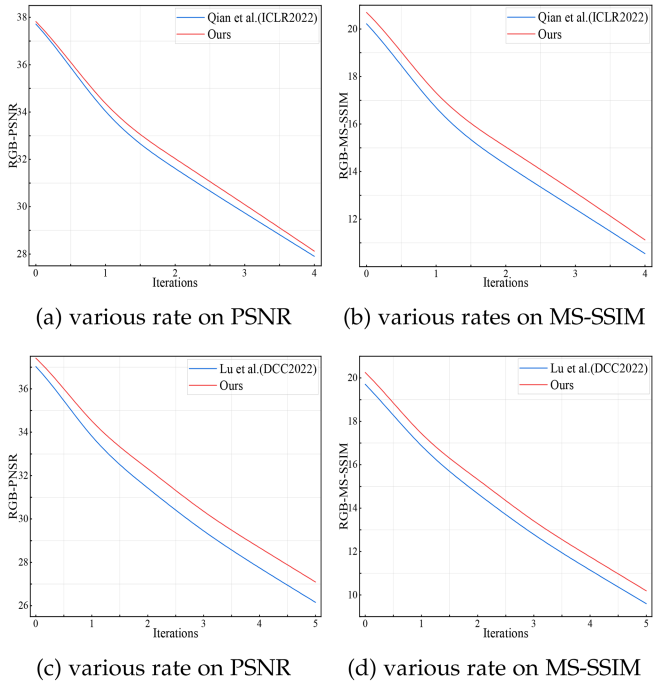



Fig. 6. Multiple continuous re-encodings experiments comparing with recent Transformer-based single-rate image compression methods on the Kodak [19] dataset. (a) and (b) Qian et al. [54], (c) and (d) Lu et al. [56] on metrics of PSNR and MS-SSIM respectively. Our I2C also consistently outperforms the Transformer-based methods further showing its robustness and advantages.

TABLE I
FIXED-RATE RE-ENCODING EVALUATION ON CLIC DATASET. THE AUC IS THE AREA UNDER THE "ITERATIONS"-"PNSR"/"MS-SSIM" CURVES. 32 RE-ENCODINGS WITH A FIXED RATE (BPP=0.318) ARE ADOPTED. THE SUPERIOR RESULTS ON FIXED RATES SHOW THAT THE PROPOSED IAT MODULE SURPRISINGLY ENHANCES THE FIDELITY MAINTENANCE ABILITY OF THE BASELINE [18]

| Method | PSNR(AUC)↑ | MS-SSIM(AUC)↑ |
|---|---|---|
| Song et al. [1] | 865.287 | 388.634 |
| Baseline [18] | 856.338 | 410.114 |
| Ours | **967.805** | **419.554** |

single-rate methods (Ballé et al. [10], Minnen et al. [29], Chen et al. [36], and Hu et al. [32]) adopt different parameter models for the different compression rates. For each comparison, we adjust the bpp of our I2C to adapt those methods.

We further construct re-encoding experiments with more (expanding from 31 to 91) iteration operations. The visualization results (up to 91 re-encodings with different compression rates) are shown in Fig. 9. Fig. 10(a) and (b) show the performance after multiple continuous operations of compression/decompression with different compression rates on the Kodak [19] dataset. The experimental results further indicate that our proposed I2C is more powerful in maintaining image fidelity with increasing re-encoding operations (I2C maintains great image fidelity even 91 iterations are executed).

Besides, we conduct experiments on biomedical and remote sensing images to show the superiority of I2C for practical images of different domains. Fig. 11 shows the qualitative results after different numbers of compression/decompression operations under various rates compared with the state-of-the-art VAE-based approach [1]. Fig. 12 is the visualization results showing those single-rate image compression methods compared with our I2C during the variable-rate re-encodings. Consistently, I2C outperforms those methods by a large margin and preserves the image fidelity much better, especially after multiple continuous re-encodings.

The results indicate that our proposed I2C is powerful to preserve image fidelity, which is important for practical applications such as media online sharing and cooperative media processing.

### C. Rate-Distortion Performance

To verify the general validity of the proposed approach, we conduct rate-distortion (RD) performance experiments on three datasets, i.e., Kodak [19], CLIC [20], and DIV2K [21]. We compare our approach with seven recent state-of-the-art learning-based image compression methods [1], [18], [32], [33], [54], [56], [72], [73], [74] and three classical codec methods, BPG [5], AVIF [6], and VVC [7]. The results of learning-based methods are collected from their official GitHub pages or their papers. The VCC approach is implemented by the official Test Model VTM 12.1 with the intra-profile configuration from the official GitHub page. Both VVC and BPG software were configured with the YUV444 format to maximize compression performance. AVIF [6] approach is implemented by the official GitHub page. We configure the AVIF software with PNG format for input to maximize compression performance.

All comparable results are demonstrated in Fig. 13. It is seen that our approach achieves the best results with commonly used metrics PSNR and MS-SSIM on three datasets. Compared with the baseline method [18], our approach achieves comparable R-D performance on the Kodak dataset [19] (Fig. 13(a), (d)) and outperforms the baseline on both the CLIC dataset [20] (Fig. 13(b), (e)) and the DIV2K dataset [21] (Fig. 13(c), (f)). This means that our approach achieves the variable-rate adaptation based on the single-rate method [18] without sacrificing any performance, verifying the effectiveness of the I2C. It is worth noting that the CLIC dataset and DIV2K dataset are high-resolution images, implying that our method is more effective on high-resolution images. Our approach empowers the network model with variable rate in addition to improving the algorithmic performance of the original model. In Fig. 13(a), (d), the test dataset is Kodak, which contains images with a resolution of 768 $\times$ 512, and it is smaller compared to CLIC and DIV2K. As image resolution decreases, feature maps fed into the IAT module become smaller and more susceptible to quantization. That is, from experimental results, our method is especially effective on high-resolution images. In Fig.13(b), the advantages obtained by our method can be seen. The overlapping part of the curve shows that performances of fixed compression rate methods are similar to ours at specific compression rates. In summary, for a fixed rate compression, once a low-resolution image is fed, the performance of our method is competitive compared to the baseline. In contrast, once a high-resolution image is fed, our proposed method can even outperform the baseline. Besides, our method still can outperform other methods no matter the

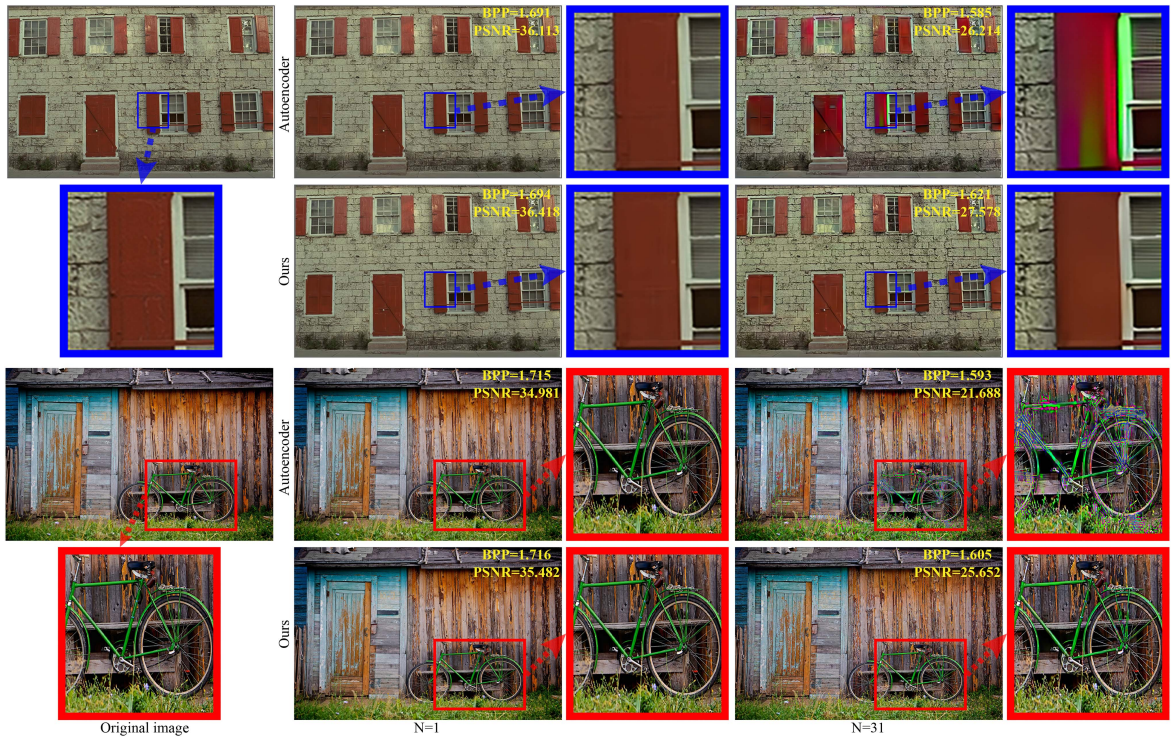



Fig. 7. Qualitative results after different numbers of compression/decompression operations under various rates. The two images (kodim1.png and alexander-shustov-73.png) are from the Kodak dataset and the CLIC dataset, respectively. Severe artifacts and color shifts would appear in the state-of-the-art VAE-based approach [1] once multiple continuous operations are executed, in contrast to better fidelity preserving of our approach. N indicates the number of compression/decompression operations. Best viewed in color.

Fig. 8. Visualization of sample images in Kodak dataset. Compared with single-rate methods (Ballé et al. [10], Minnen et al. [29], Chen et al. [36], and Hu et al. [32]) which need multiple models to achieve different compression rates. For each comparison, we adjust the bpp of our I2C to adapt those methods. Our I2C has much better fidelity-preserving performance and is a variable-rate method.

TABLE II
AREA UNDER CURVE (AUC) OF OUR METHOD AND XIE ET AL. [18] (BASELINE) ON DIFFERENT DATASETS ABOUT PSNR AND MS-SSIM. THE BPP RANGE IS DETERMINED BY THE INTERSECTION OF TWO METHODS. OUR APPROACH MAKES A SINGLE-RATE BASELINE COMPRESSION MODEL ACHIEVE THE VARIABLE-RATE ABILITY AND EVEN OUTPERFORMS THE BASELINE IN R-D PERFORMANCE

| Dataset | Xie et al. [18] | | Ours | |
|---|---|---|---|---|
| | $AUC_{PSNR}$ | $AUC_{MS\text{-}SSIM}$ | $AUC_{PSNR}$ | $AUC_{MS\text{-}SSIM}$ |
| Kodak | 32.7866 | 16.5030 | **32.7883** | **16.5036** |
| CLIC | 23.5896 | 11.7463 | **23.7082** | **11.8571** |
| DIV2K | 28.0998 | 14.7868 | **28.2138** | **14.8901** |

TABLE III
VARIABLE-RATE CONTROL EXPERIMENTS OVER THE KODAK DATASET. OUR APPROACH CAN FINELY CONTROL THE COMPRESSION RATE WITHIN THE WHOLE BPP RANGE (NO MATTER LOW OR HIGH)

| LOW | | | HIGH | | |
|---|---|---|---|---|---|
| BPP | PSNR(dB) | MS-SSIM(dB) | BPP | PSNR(dB) | MS-SSIM(dB) |
| 0.28181 | 31.6951 | 14.5015 | 1.02433 | 38.3226 | 21.2580 |
| 0.28265 | 31.7071 | 14.5153 | 1.02587 | 38.3312 | 21.2664 |
| 0.28342 | 31.7177 | 14.5263 | 1.02733 | 38.3388 | 21.2717 |
| 0.28416 | 31.7291 | 14.5377 | 1.02910 | 38.3468 | 21.2819 |
| 0.28500 | 31.7435 | 14.5517 | 1.03071 | 38.3548 | 21.2903 |
| 0.28576 | 31.7538 | 14.5639 | 1.03250 | 38.3625 | 21.2995 |
| 0.28659 | 31.7657 | 14.5765 | 1.03406 | 38.3703 | 21.3087 |
| 0.28734 | 31.7761 | 14.5874 | 1.03564 | 38.3767 | 21.3190 |
| 0.28808 | 31.7884 | 14.5952 | 1.03733 | 38.3872 | 21.3291 |
| 0.28880 | 31.8004 | 14.6092 | 1.03885 | 38.3943 | 21.3355 |

resolution of the input image. To further compare the performance between the baseline [18] and our method, we calculate their corresponding area under curve (AUC) values, as shown in Table II. The results show that our approach outperforms the single-rate model method by Xie et al. [18] in terms of the aggregated AUC metric.

In addition, Our I2C could achieve variable-rate image compression with fine granularity. To verify the effectiveness of fine variable-rate control, we illustrate multiple performances of fine variable-rate control within the low and high bpp range in Table III. In practice, classical image codecs provide hundreds of variable-rate RD points to meet the basic requirements of the application. Compared with that, our method obtains at least


Fig. 9. Qualitative results after different numbers of re-encoding operations (expanding from 31 to 91) under various rates. Severe artifacts and color shifts would appear in the Song et al. [1] once multiple continuous operations are executed.

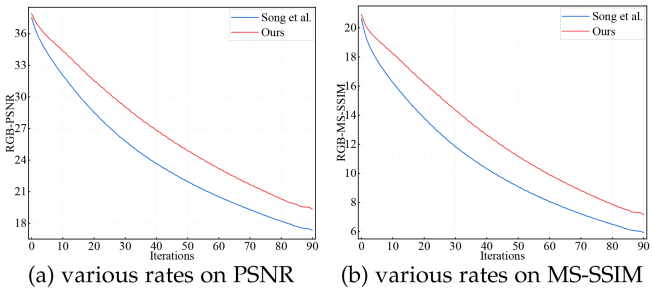



Fig. 10. Up to 91 successive re-encodings on the Kodak [19] dataset. The compression rates of each re-encoding operation are different. I2C outperforms Song et al. [1] by a large margin to show the superiority of fidelity preserving especially when multiple continuous operations are executed.

1000 effective variable-rate RD points with a very fine PSNR and MS-SSIM.

### D. Reduction in the Volume of Model Parameters

To further improve the efficiency of our model, we try to additionally reduce its parameters without performance compromise. Compared to the original version shown in Fig. 2, the number of channels of feature maps, which are before and after channel averaging operations, is reduced from 768 to 192. Therefore, we then consider inserting the IAT module after the channel averaging operation instead of before it to make the entire model more lightweight. Also, due to the ability of latent representation modeling of I2C, we further remove the attention module of the I2C RealNVP-based version to reduce the number of parameters. The experiments show that such a simplification has no significant impact on the performance of the algorithm. Fig. 14 shows the network framework of the lightweight version. We present the computational costs and model size of Ours and $\text{Ours}_{\text{light}}$ in Table IV, the number of parameters has been reduced by nearly one-third, and the reduction of the number of parameters has taken a step toward the practicability of the algorithm deployment. We tested the algorithm performance on three datasets: Kodak [19], CLIC Professional Validation [20] and DIV2K validation [21]. Since the RD-performance curves are close, we calculated the AUC (the higher the value, the better the performance) to be able to see more intuitively the algorithm performance comparison before and after the parametric number change. The results of the experiment are shown in Table V, the number of parameters is reduced by nearly one-third, but the performance of the algorithm is not degraded, which shows that this simplification is effective.

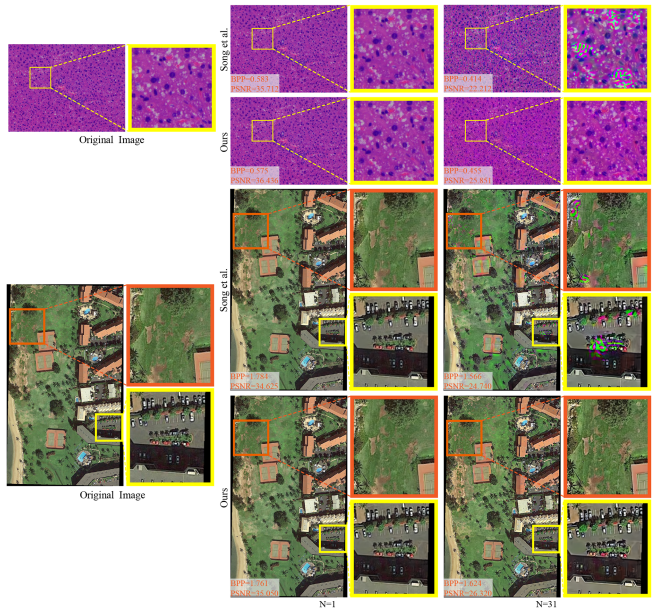



Fig. 11. Qualitative results after different numbers of compression/decompression operations under various rates on practical biomedical and remote sensing images. A similar conclusion of Fig. 7 can be achieved that severe artifacts and color shifts would appear in the state-of-the-art VAE-based approach [1] once multiple continuous operations are executed, in contrast to better fidelity preserving of our approach. N indicates the number of compression/decompression operations. Best viewed in color.

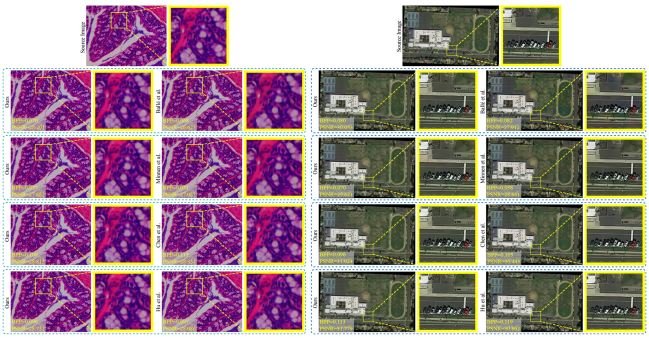


Fig. 12. Visualization of samples on practical biomedical and remote sensing images. Compared with single-rate methods (Ballé et al. [10], Minnen et al. [29], Chen et al. [36], and Hu et al. [32]) which need multiple models to achieve different compression rates. Same as the settings of Fig. 8, for each comparison, we adjust the bpp of our I2C to adapt those methods. Consistently, our I2C has also much better RD performance on practical images of different domains.

TABLE IV
COMPARE THE PARAMETER AND COMPUTATIONAL COST AMONG OURS, OUR LIGHTWEIGHT VERSION, BASELINE [18], AND ANOTHER TWO TYPICAL LEARNING-BASED APPROACHES ([54], [56]). THE GFLOPS AND RUNNING MEMORY ARE OBTAINED BY INPUTTING THE 512*768 RGB IMAGE. N DENOTES THE NUMBER OF DIFFERENT COMPRESSION RATE MODELS. IT IS WORTH NOTING THAT OUR PROPOSED I2C HAS MORE THAN 1000 VARIABLE COMPRESSION RATES, I.E., $N > 1000$, IN ADDITION TO MAINTAINING HIGH FIDELITY CONTINUOUS CODEC

| Method | GFLOPs | Parameters (M) | Memory (GB) | Training cost (Days) |
|---|---|---|---|---|
| Lu et al. [56] | 216.834 | $15.952 \times N$ | 3.593 | $4 \times N$ |
| Qian et al. [54] | 180.258 | $42.686 \times N$ | 3.532 | $7 \times N$ |
| Baseline [18] | 379.329 | $45.345 \times N$ | 2.492 | $10 \times N$ |
| $\text{Ours}_{\text{light}}$ | 403.854 | $44.904 \times 1$ | 2.558 | $16 \times 1$ |
| Ours | 472.491 | $68.513 \times 1$ | 2.665 | $18 \times 1$ |

TABLE V
AREA UNDER CURVE (AUC) OF OURS AND OUR LIGHTWEIGHT VERSION ON THREE DIFFERENT DATASETS OF PSNR AND MS-SSIM. THE BPP RANGE IS DETERMINED BY THE INTERSECTION OF TWO METHODS. THE LIGHTWEIGHT VERSION KEEPS A COMPETITIVE R-D PERFORMANCE COMPARED TO THE ORIGINAL ONE

| Dataset | Ours | | $\text{Ours}_{\text{light}}$ | |
|---|---|---|---|---|
| | $\text{AUC}_{\text{PSNR}}$ | $\text{AUC}_{\text{MS-SSIM}}$ | $\text{AUC}_{\text{PSNR}}$ | $\text{AUC}_{\text{MS-SSIM}}$ |
| Kodak | 32.8444 | 16.5020 | 32.8408 | 16.4993 |
| CLIC | 23.8422 | 11.9030 | 23.8401 | 11.9054 |
| DIV2K | 28.3760 | 14.9412 | 28.3763 | 14.9438 |

### E. The Complexity of I2C

In this subsection, we present the complexity of our proposed I2C about the size of parameters, GFLOPs, running memory, and train costs. The statistics of GFLOPs and running memory are performed during the inference procedure, and the ANS entropy coding [66] (adopted by all methods in the same way, running on CPU) is not included to facilitate the statistics. The proposed I2C outperforms the baseline [18] not only in rate-distortion performance but also in efficiency. Since the proposed model could obtain variable-rate image compression within a single model, we can reduce a large amount of additional training and storing once different compression rates are required in a task. The baseline [18] and another two typical learning-based methods ([54], [56]) take about 10/7/4 days to train a fixed-rate model on one single Nvidia RTX 3090 GPU respectively. However, once N different compression rates are required, the training time and storing cost would heavily increase to N times. With the same computational environment, our proposed I2C only requires 18 days (or 16 days for the lightweight version) to train and could achieve more than 1000 different compression rates, as shown in Table IV.

### F. I2C With Different INN-Based Architectures

We use three different coupling layers from Incompressible-flow Network (GIN) [23], NICE [24], and GLOW [25] instead of the affine coupling layer, initially proposed in RealNVP [26] and adopted by the invertible block of baseline model [18] in Fig. 2, to verify that I2C can adapt to different INN-based architectures. We conduct RD performance experiments on three datasets, i.e., Kodak [19], CLIC [20], and DIV2K [21]. In order to compare the performance with $\text{Our}_{\text{NICE}}$, $\text{Our}_{\text{GIN}}$, and $\text{Ours}_{\text{GLOW}}$ methods, we calculate their corresponding area under curve (AUC) values, as shown in Table VI. Since the GIN [23] preserves volumes of the INN and the Jacobian determinant is simply unity, the result is better lightly than our I2C RealNVP-based version. It can be seen from the experimental results that I2C can be readily applied to different INN-based architectures.

## VI. DISCUSSION

### A. Codec Processing of Variable-Rate Control

It is worth noting that there are three crucial differences in achieving the fine variable-rate control between Song et al. [1] and ours. First, the input of controlling is different. The tensor-based Lagrange multiplier is computed by the quality level, which is different from the quality map input of Song et al. [1]. The quality map of Song et al. [1] represents the semantic segmentation map for task-aware image compression, and our quality level represents the compression level. Second, the additional information is different. The training process is different because our quality level is different from the quality map of Song et al. [1], which requires semantic segmentation labels, and we do not need semantic segmentation labels, enabling more flexible image compression. Our method is able to train on arbitrary images without semantic segmentation labels, as we mention in Section III-C. Finally, The codec processing of variable-rate controlling is different. Song et al. [1] used different quality maps in the encoding and decoding process, i.e., the quality map in the decoding process is generated by the latent representation through neural networks, which is not equal to the input quality map in the encoding process. Differently, we control the variable-rate image compression by storing the quality level in the bitstream directly. The usage of the same quality level of both encoding and decoding procedures would bring out a more stable and finer controlling result. To validate this idea, we conduct fine variable-rate control comparison experiments with 10000 points. We implemented Song et al. [1] according to their official GitHub code. It can be seen in Table VII that with the bpp increases, Song et al. [1] show occasional decay in PSNR and MS-SSIM, while ours are consistently increasing.

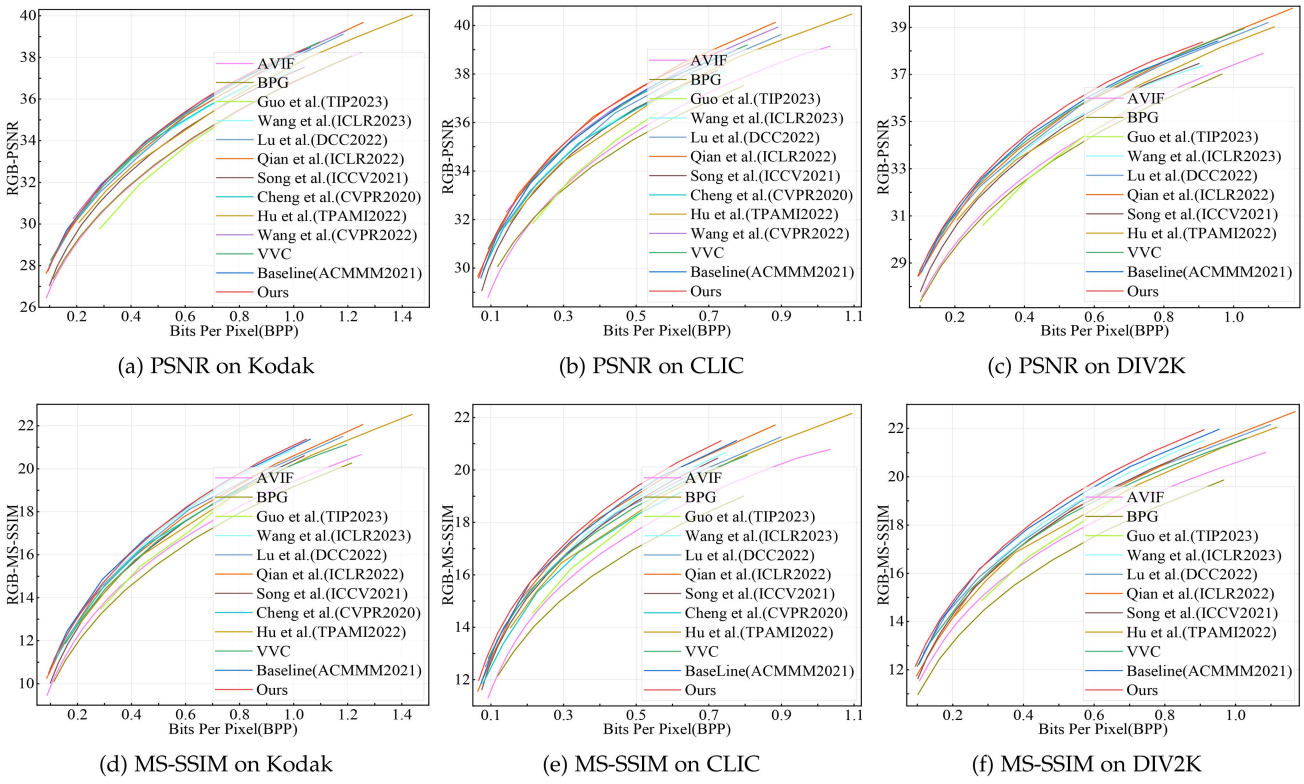



Fig. 13. RD performance curves aggregated over the Kodak [19], CLIC professional validation dataset [20], and DIV2K validation dataset [21]. MS-SSIM values converted to decibels ($-10log_{10}(1-MS\text{-}SSIM)$). (a)–(c) and (d)–(f) are results on Kodak, CLIC, and DIV2K about PSNR and MS-SSIM, respectively. It is worth noting that CLIC and DIV2K are datasets with high-resolution images. That is, our method is especially effective on high-resolution images.

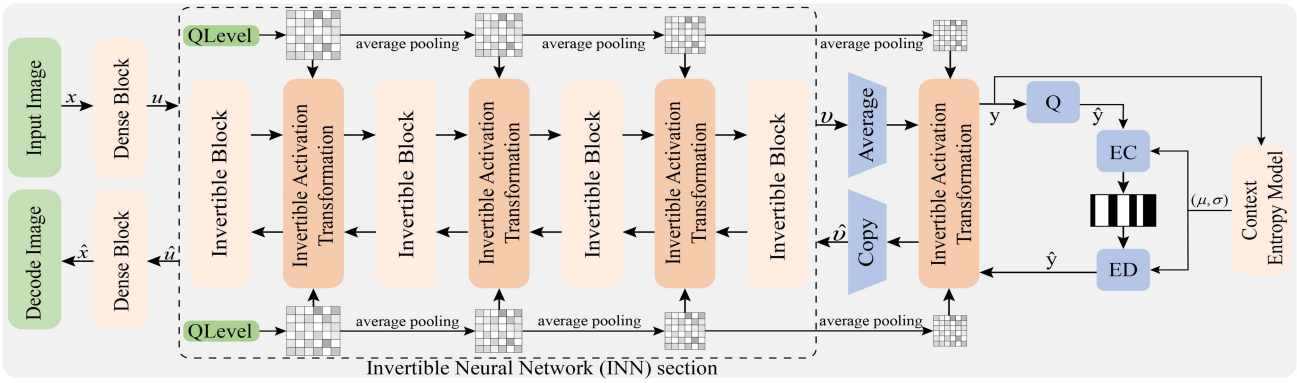


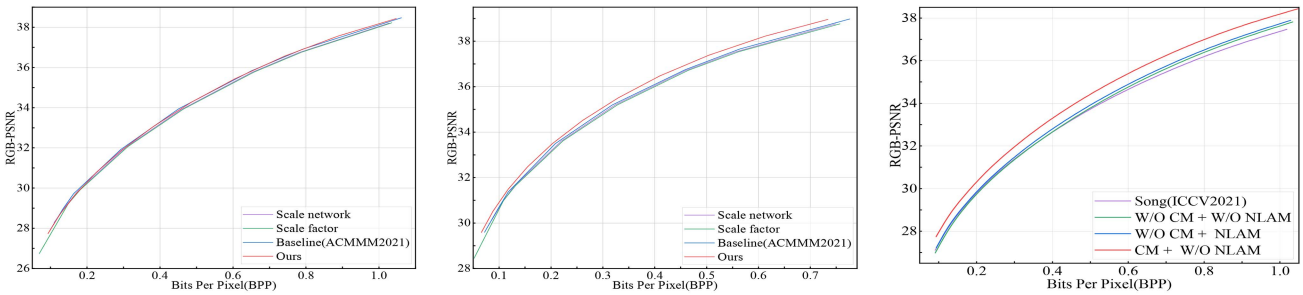

Fig. 14. Lightweight version of network architecture equipped with the proposed Invertible Activation Transformation (IAT) module. The last IAT module inserts after the channel squeeze operation to reduce the volume of the model parameters. EC/ED means entropy encoding/decoding respectively. Q is the quantizer.

TABLE VI
AREA UNDER CURVE (AUC) OF OURS, $\text{OURS}_{\text{GIN}}$, $\text{OURS}_{\text{NICE}}$ AND $\text{OURS}_{\text{GLOW}}$ ON DIFFERENT DATASETS OF PSNR AND MS-SSIM. THE BPP RANGE IS DETERMINED BY THE INTERSECTION OF FOUR METHODS. IT CAN BE SEEN FROM THE EXPERIMENTAL RESULTS THAT I2C CAN BE READILY APPLIED TO DIFFERENT INN-BASED ARCHITECTURES

| Dataset | Ours | | $\text{Ours}_{\text{GIN}}$ | | $\text{Ours}_{\text{NICE}}$ | | $\text{Ours}_{\text{GLOW}}$ | |
|---|---|---|---|---|---|---|---|---|
| | $\text{AUC}_{\text{PSNR}}$ | $\text{AUC}_{\text{MS-SSIM}}$ | $\text{AUC}_{\text{PSNR}}$ | $\text{AUC}_{\text{MS-SSIM}}$ | $\text{AUC}_{\text{PSNR}}$ | $\text{AUC}_{\text{MS-SSIM}}$ | $\text{AUC}_{\text{PSNR}}$ | $\text{AUC}_{\text{MS-SSIM}}$ |
| Kodak | 32.4607 | 16.3008 | 32.5049 | 16.3115 | 32.3582 | 16.2012 | 32.5253 | 16.2651 |
| CLIC | 23.5862 | 11.7673 | 23.6004 | 11.7484 | 23.5164 | 11.7181 | 23.5889 | 11.6833 |
| DIV2K | 28.0639 | 14.7713 | 28.0944 | 14.7635 | 27.9752 | 14.6985 | 28.1108 | 14.7187 |

TABLE VII
COMPARISON WITH SONG ET AL. [1] ON BPP FINE VARIABLE-RATE CONTROLLING. IDEALLY, AS BPP INCREASES, PSNR AND MS-SSIM SHOULD ALSO INCREASE ACCORDINGLY. HOWEVER, PSNR AND MS-SSIM OF SONG ET AL. [1] DO NOT KEEP CONSISTENTLY INCREASING WHILE OCCASIONAL DECAYS OCCUR, INDICATING THAT OUR PROPOSED I2C ACHIEVES FINER AND MORE STABLE VARIABLE-RATE CONTROLLING DUE TO DIFFERENT INPUTS AND STRATEGIES

| Ours | | | | | Song et al. [1] | | | | |
|---|---|---|---|---|---|---|---|---|---|
| BPP | PSNR | PSNR difference value | MS-SSIM | MS-SSIM difference value | BPP | PSNR | PSNR difference value | MS-SSIM | MS-SSIM difference value |
| 0.108585 | 27.716038 | / | 10.669252 | / | 0.108582 | 27.334372 | / | 10.328648 | / |
| 0.108605 | 27.717019 | 0.000981 | 10.670246 | 0.000994 | 0.108605 | 27.334668 | 0.000296 | 10.328349 | **-0.000299** |
| 0.108636 | 27.717729 | 0.000710 | 10.670953 | 0.000707 | 0.108626 | 27.336041 | 0.001372 | 10.329235 | 0.000886 |
| 0.108663 | 27.718516 | 0.000786 | 10.671773 | 0.000820 | 0.108632 | 27.329554 | **-0.006487** | 10.329636 | 0.000401 |
| 0.108707 | 27.719769 | 0.001253 | 10.672980 | 0.001207 | 0.108670 | 27.328086 | **-0.001468** | 10.329518 | **-0.000119** |
| 0.108734 | 27.720084 | 0.000315 | 10.674001 | 0.001021 | 0.108673 | 27.329059 | 0.000973 | 10.330991 | 0.001474 |
| 0.108775 | 27.721330 | 0.001246 | 10.675116 | 0.001115 | 0.108707 | 27.329655 | 0.000596 | 10.331930 | 0.000938 |
| 0.108795 | 27.722693 | 0.001363 | 10.675595 | 0.000479 | 0.108721 | 27.330576 | 0.000921 | 10.332789 | 0.000860 |
| 0.108819 | 27.723348 | 0.000655 | 10.675724 | 0.000129 | 0.108761 | 27.331321 | 0.000745 | 10.332517 | **-0.000272** |
| 0.108856 | 27.724760 | 0.001412 | 10.677004 | 0.001281 | 0.108792 | 27.332157 | 0.000836 | 10.333420 | 0.000903 |
| 0.108894 | 27.726053 | 0.001293 | 10.678917 | 0.001912 | 0.108802 | 27.329894 | **-0.002264** | 10.332498 | **-0.000922** |
| 0.108907 | 27.727370 | 0.001317 | 10.679967 | 0.001050 | 0.108832 | 27.331151 | 0.001258 | 10.333511 | 0.001013 |




(a) PSNR on Kodak (b) PSNR on CLIC (c) Impact of Gain Components.

Fig. 15. (a) and (b) represent the impact of the QLevel representation. The scale factor method (green line) is similar to Chen et al. [13]. The scale network method (purple line) is similar to Lin et al. [16]. Our proposed tensor-based QLevel representation achieves better performance than simply using a scalar or scale network to control the compression rate. (c) represents the impact of gain components. W/O represents "without", W represents "with", CM represents "context model", and NLAM represents "non-local attention module".

The result indicates that our fine variable-rate control is different from Song et al. [1], and shows better and more stable results on fine variable-rate control. In addition, the quality level is a uniform tensor generated from a single value during the testing. The number of stored bits in the bitstream depends on the granularity of desirable variable-rate control. Theoretically, if 8 bits are used, $2^8 = 256$ effective variable rates are achieved. If 16 bits are used, $2^{16} = 65536$ effective variable rates are achieved. This small amount of additional bitstream storage can generate a large number of fine variable rates, which is quite practical in real-world applications.

### B. Impact of the QLevel Representation

To further analyze the effectiveness of the tensor-based QLevel representation of our proposed I2C, we conducted an ablation study by modifying the quality level representation. We compared the proposed approach with the baseline method [18] and the simplified version of our method, which modifies the quality level from tensor to scalar, similar to [13]. We also conduct the comparison experiment with the method of Lin et al. [16], which used the scaling network (Scale network) to map the scalar value of the Lagrange multiplier into a vector channel-wisely scale feature map achieving the variable compression rate. It is worth noting that the method of Lin et al. [16] does not satisfy mathematical invertibility and cannot be used directly, so we modified it to be applicable to the invertible neural network-based architecture. Comparative results are shown in Fig. 15(a) and (b). The results indicate that the proposed tensor-based quality level can obtain better performance, compared with the scalar factor one, which only provides channel-wise weighted computations on latent representation. I2C achieves the same great advantage compared to the Scale network one (Lin et al. [16]).

### C. Impact of Gain Components

The context model [27], [28], [29] and the non-local attention module [36] are commonly used in the learned-based image compression methods to further reduce statistical redundancy within the latent features and improve the probabilistic estimation ability of the network. We conduct an ablation study to evaluate the impact of the context model and non-local attention module on our method in the Kodak dataset [19], as shown in Fig. 15(c). We start from a baseline without the context model and non-local attention module, i.e., W/O CM (context model) and W/O NLAM (non-local attention module), and plot the rate-distortion performance in green color. Then, we add the non-local attention module (blue color) and context model (red

color) to evaluate the performance. We can observe that using the context model achieves the best results, while it requires high computational costs (codec process takes about 233 seconds on an Intel (R) Core (TM) i9-10900 K CPU on Kodak and includes entropy encoding/decoding procedure). Once the context model is removed, I2C could be implemented on GPU platforms in a parallel computing manner and the codec time would reduce to 5.694 seconds on one NVIDIA RTX 3090 GPU. In addition, even if the context model and non-local attention module are removed from I2C, our method still outperforms Song et al. [1], demonstrating the effectiveness of the proposed method.

### D. The Suitability of INN for High-Fidelity Codec

To investigate the reason why our I2C could better handle the problem, we further conduct the error accumulation analysis during the re-encoding procedure. In the lossy image compression procedure of re-encodings, the error accumulation can be described in the following formulation:

$$\Delta = \underbrace{\Delta_{\mathrm{e}}^{1} \circ \Delta_{\mathrm{q}}^{1} \circ \Delta_{\mathrm{d}}^{1} \circ \Delta_{\mathrm{rc}}^{1}}_{\text{Iteration 1}} \circ \underbrace{\Delta_{\mathrm{e}}^{2} \circ \Delta_{\mathrm{q}}^{2} \circ \Delta_{\mathrm{d}}^{2} \circ \Delta_{\mathrm{rc}}^{2}}_{\text{Iteration 2}} \circ \cdots \circ \underbrace{\Delta_{\mathrm{e}}^{\mathrm{N}} \circ \Delta_{\mathrm{q}}^{\mathrm{N}} \circ \Delta_{\mathrm{d}}^{\mathrm{N}} \circ \Delta_{\mathrm{rc}}^{\mathrm{N}}}_{\text{Iteration N}}, \tag{19}$$

where $\circ$ denotes the function composition. During the n-th step of the re-encoding iteration, $\Delta_{\mathrm{q}}^{\mathrm{n}}$, $\Delta_{\mathrm{e}}^{\mathrm{n}}$, $\Delta_{\mathrm{d}}^{\mathrm{n}}$, and $\Delta_{\mathrm{rc}}^{\mathrm{n}}$ denote error of quantization, error of encoding transformation, error of decoding transformation, and error of rounding and clipping, respectively.

In our proposed I2C, errors of encoding and decoding transformation ($\Delta_{\mathrm{e}}^{\mathrm{n}}$ and $\Delta_{\mathrm{d}}^{\mathrm{n}}$) are composed of small numbers of nonlinear layers (e.g., dense block) and plenty of bijective mapping layers. The error accumulation can be expressed as the following:

$$\Delta_{\text{ed-ours}}^{\mathrm{n}} = \Delta_{\mathrm{e}}^{\mathrm{n}} \circ \Delta_{\mathrm{q}}^{\mathrm{n}} = \Delta_{\text{nonlinear-1}}^{\mathrm{n}} \circ \Delta_{\text{nonlinear-2}}^{\mathrm{n}} \circ \Delta_{\text{bijective}}^{\mathrm{n}}. \tag{20}$$

In the VAE-based method (e.g., [1]), errors of encoding and decoding transformation ($\Delta_{\mathrm{e}}^{\mathrm{n}}$ and $\Delta_{\mathrm{d}}^{\mathrm{n}}$) are composed of massive nonlinear layers (e.g., resnet block). The error accumulation can be expressed as the following:

$$\Delta_{\text{ed-VAE}}^{\mathrm{n}} = \Delta_{\mathrm{e}}^{\mathrm{n}} \circ \Delta_{\mathrm{q}}^{\mathrm{n}} = \Delta_{\text{nonlinear-1}}^{\mathrm{n}} \circ \Delta_{\text{nonlinear-2}}^{\mathrm{n}} \circ \cdots \circ \Delta_{\text{nonlinear-T}}^{\mathrm{n}}. \tag{21}$$

The architecture of our proposed I2C is primarily composed of bijective mapping layers, which exhibit a mathematical invertible property to avoid discarding any information in the latent space, resulting in preserving high fidelity. The error accumulation in the encoding and decoding transformation of our proposed I2C is much less than the VAE-based approach ($\Delta_{\text{ed-ours}}^{\mathrm{n}} < \Delta_{\text{ed-VAE}}^{\mathrm{n}}$) (especially with the growing complexity of the model, our I2C has a fixed small number of nonlinear layers while the number of nonlinear layers T of the VAE-based method is increasing). Therefore, our proposed I2C can achieve the high fidelity of reconstructed images in the continuous codec process more efficiently.

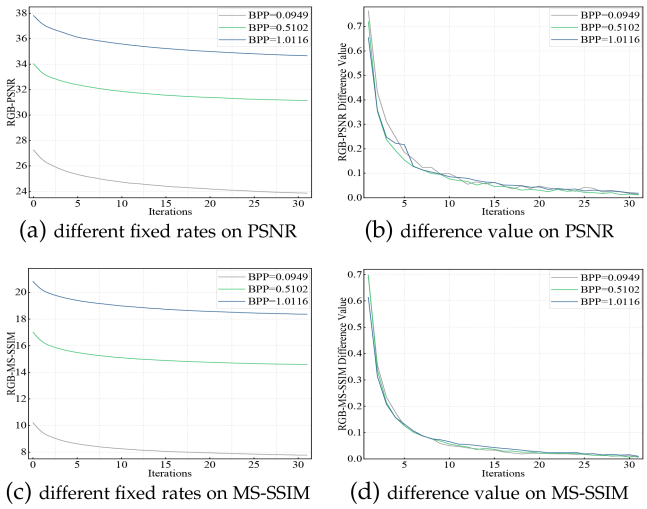



Fig. 16. Successive re-encodings on the Kodak [19] dataset with different fixed rates. (a), (b) and (c), (d) are the results of the PSNR/PSNR Difference Value and MS-SSIM/MS-SSIM Difference Value, respectively.

### E. Impact of Re-encodings with Different Fixed Rates

We analyze the effect of re-encodings with different fixed compression rates ($\lambda s$) at low, medium, and high bpp, respectively. Fig. 16(a), (c) illustrates the decay of PSNR/MS-SSIM with increasing re-encoding operations at three different rates. Fig. 16(b), (d) shows the gradual decrease of the PSNR/MS-SSIM difference value between two adjacent re-encoding operations as the number of iterations increases. The reconstructed image quality will be better at higher bits-per-pixel (BPP), as shown in Fig. 16(a), (c). Besides, we also find that the decay tendencies of the PSNR/MS-SSIM remain consistent for different compression rates during re-encoding operations. As discussed in Section VI-D, the accumulation of errors comes from four aspects (refer to (19)), our proposed I2C greatly eliminates the errors ($\Delta_{\mathrm{e}}^{\mathrm{n}}$ and $\Delta_{\mathrm{d}}^{\mathrm{n}}$) generated by the encoding/decoding transformation, since the number of nonlinear layers of our I2C is small and fixed no matter the compression rate is. Thus given different fixed $\lambda$s, the system would gradually converge to a stable state as the number of iterations increases, as verified by Fig. 16(b), (d).

## VII. Conclusion

In this paper, we propose a high-fidelity variable-rate image compression method by introducing the Invertible Continuous Codec (I2C). We construct the I2C based on Invertible Neural Network (INN) with the core Invertible Activation Transformation (IAT) module implemented in a mathematical invertible manner. IAT is actually a feature activation transform layer of the INN and has the ability of fine variable-rate control by feeding the quality level (QLevel) to generate the scaling and bias tensors while better preserving the image fidelity. Extensive experiments demonstrate that thanks to the invertible design of I2C, fewer artifacts or color shifts would have appeared and the fidelity of reconstructed images is better preserved, especially when

multiple continuous re-encodings are executed under various compression rates. I2C is also able to achieve fine variable-rate control without any performance compromise.

## ACKNOWLEDGMENT

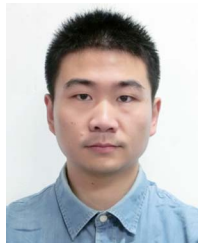
The computation is completed in the HPC Platform of Huazhong University of Science and Technology.

## REFERENCES

[1] M. Song, J. Choi, and B. Han, "Variable-rate deep image compression through spatially-adaptive feature transform," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2021, pp. 2360–2369.
[2] G. K. Wallace, "The JPEG still picture compression standard," *IEEE Trans. Consum. Electron.*, vol. 38, no. 1, pp. 18–34, Feb. 1992.
[3] M. Rabbani, "JPEG2000: Image compression fundamentals, standards and practice," *J. Electron. Imag.*, vol. 11, no. 2, 2002, Art. no. 286.
[4] Google, "Web picture format," 2010. [Online]. Available: https://chromium.googlesource.com/webm/libwebp
[5] F. Bellard, "BPG image format," 2015. [Online]. Available: https://bellard.org/bpg/
[6] AOMedia, "AV1 image file format (AVIF)," 2022. [Online]. Available: https://github.com/AOMediaCodec/libavif/releases/tag/v1.0.1
[7] J. V. E. T. (JVET), "VVC official test model VTM," 2021, Accessed: Apr. 5, 2021. [Online]. Available: https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware_VTM/-/tree/VTM-12.1
[8] J. Ballé, V. Laparra, and E. P. Simoncelli, "End-to-end optimized image compression," in *Proc. Int. Conf. Learn. Representations*, 2017.
[9] L. Theis, W. Shi, A. Cunningham, and F. Huszár, "Lossy image compression with compressive autoencoders," in *Proc. Int. Conf. Learn. Representations*, 2017.
[10] J. Ballé, D. Minnen, S. Singh, S. J. Hwang, and N. Johnston, "Variational image compression with a scale hyperprior," in *Proc. Int. Conf. Learn. Representations*, 2018.
[11] Y. Choi, M. El-Khamy, and J. Lee, "Variable rate deep image compression with a conditional autoencoder," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 3146–3154.
[12] F. Yang, L. Herranz, J. V. D. Weijer, J. A. I. Guitián, A. M. López, and M. G. Mozerov, "Variable rate deep image compression with modulated autoencoder," *IEEE Signal Process. Lett.*, vol. 27, pp. 331–335, 2020.
[13] T. Chen and Z. Ma, "Variable bitrate image compression with quality scaling factors," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2020, pp. 2163–2167.
[14] Z. Sun et al., "Interpolation variable rate image compression," in *Proc. 29th ACM Int. Conf. Multimedia*, 2021, pp. 5574–5582.
[15] Z. Cui, J. Wang, S. Gao, T. Guo, Y. Feng, and B. Bai, "Asymmetric gained deep image compression with continuous rate adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 10 527–10 536.
[16] J. Lin et al., "Variable-rate multi-frequency image compression using modulated generalized octave convolution," in *Proc. 22nd Int. Multimedia Signal Process. Workshops*, 2020, pp. 1–6.
[17] L. Helminger, A. Djelouah, M. Gross, and C. Schroers, "Lossy image compression with normalizing flows," in *Proc. Int. Conf. Learn. Representations Workshops*, 2021.
[18] Y. Xie, K. L. Cheng, and Q. Chen, "Enhanced invertible encoding for learned image compression," in *Proc. 29th ACM Int. Conf. Multimedia*, 2021, pp. 162–170.
[19] E. K. Company, "Kodak lossless true color image suite," 1999. [Online]. Available: http://r0k.us/graphics/kodak/
[20] G. Toderici et al., "Workshop and challenge on learned image compression," 2020. [Online]. Available: http://www.compression.cc
[21] E. Agustsson and R. Timofte, "Ntire 2017 challenge on single image super-resolution: Dataset and study," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2017, pp. 126–135.
[22] S. Cai, Z. Zhang, L. Chen, L. Yan, S. Zhong, and X. Zou, "High-fidelity variable-rate image compression via invertible activation transformation," in *Proc. 30th ACM Int. Conf. Multimedia*, 2022, pp. 2021–2031.
[23] P. Sorrenson, C. Rother, and U. Köthe, "Disentanglement by nonlinear ICA with general incompressible-flow networks (GIN)," in *Proc. Int. Conf. Learn. Representations*, 2020.
[24] L. Dinh, D. Krueger, and Y. Bengio, "Nice: Non-linear independent components estimation," in *Proc. Int. Conf. Learn. Representations Workshops*, 2015.
[25] D. P. Kingma and P. Dhariwal, "Glow: Generative flow with invertible 1x1 convolutions," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 10 236–10 245.
[26] L. Dinh, J. Sohl-Dickstein, and S. Bengio, "Density estimation using real NVP," in *Proc. Int. Conf. Learn. Representations*, 2017.
[27] J. Lee, S. Cho, and S.-K. Beack, "Context-adaptive entropy model for end-to-end optimized image compression," in *Proc. Int. Conf. Learn. Representations*, 2019.
[28] F. Mentzer, E. Agustsson, M. Tschannen, R. Timofte, and L. V. Gool, "Conditional probability models for deep image compression," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4394–4402.
[29] D. Minnen, J. Ballé, and G. D. Toderici, "Joint autoregressive and hierarchical priors for learned image compression," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 10 794–10 803.
[30] Z. Guo, Y. Wu, R. Feng, Z. Zhang, and Z. Chen, "3-D context entropy model for improved practical image compression," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2020, pp. 520–523.
[31] Y. Hu, W. Yang, and J. Liu, "Coarse-to-fine hyper-prior modeling for learned image compression," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 11 013–11 020.
[32] Y. Hu, W. Yang, Z. Ma, and J. Liu, "Learning end-to-end lossy image compression: A benchmark," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 8, pp. 4194–4211, Aug. 2022.
[33] Z. Cheng, H. Sun, M. Takeuchi, and J. Katto, "Learned image compression with discretized Gaussian mixture likelihoods and attention modules," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 7936–7945.
[34] D. Minnen and S. Singh, "Channel-wise autoregressive entropy models for learned image compression," in *Proc. IEEE Int. Conf. Image Process.*, 2020, pp. 3339–3343.
[35] L. Zhou, Z. Sun, X. Wu, and J. Wu, "End-to-end optimized image compression with attention mechanism," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2019.
[36] T. Chen, H. Liu, Z. Ma, Q. Shen, X. Cao, and Y. Wang, "End-to-end learnt image compression via non-local attention optimization and improved context modeling," *IEEE Trans. Image Process.*, vol. 30, pp. 3179–3191, 2021.
[37] Y. Ma, Y. Zhai, J. Yang, C. Yang, and R. Wang, "AFEC: Adaptive feature extraction modules for learned image compression," in *Proc. 29th ACM Int. Conf. Multimedia*, 2021, pp. 5436–5444.
[38] Y. Mei, L. Li, Z. Li, and F. Li, "Learning-based scalable image compression with latent-feature reuse and prediction," *IEEE Trans. Multimedia*, vol. 24, pp. 4143–4157, 2022.
[39] M. Li, K. Zhang, J. Li, W. Zuo, R. Timofte, and D. Zhang, "Learning context-based nonlocal entropy modeling for image compression," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 3, pp. 1132–1145, Mar. 2023.
[40] Z. Guo, Z. Zhang, R. Feng, and Z. Chen, "Causal contextual prediction for learned image compression," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 4, pp. 2329–2341, Apr. 2022.
[41] A. Punnappurath and M. S. Brown, "Learning raw image reconstruction-aware deep image compressors," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 4, pp. 1013–1019, Apr. 2020.
[42] D. Tellez, G. Litjens, J. van der Laak, and F. Ciompi, "Neural image compression for gigapixel histopathology image analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 2, pp. 567–578, Feb. 2021.
[43] C. Cai, L. Chen, X. Zhang, and Z. Gao, "End-to-end optimized ROI image compression," *IEEE Trans. Image Process.*, vol. 29, pp. 3442–3457, 2020.
[44] H. Ma, D. Liu, N. Yan, H. Li, and F. Wu, "End-to-end optimized versatile image compression with wavelet-like transform," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 3, pp. 1247–1263, Mar. 2022.
[45] Y.-H. Ho, C.-C. Chan, W.-H. Peng, H.-M. Hang, and M. Domański, "ANFIC: Image compression using augmented normalizing flows," *IEEE Open J. Circuits Syst.*, vol. 2, pp. 613–626, 2021.
[46] O. Rippel and L. Bourdev, "Real-time adaptive image compression," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 2922–2930.
[47] E. Agustsson, M. Tschannen, F. Mentzer, R. Timofte, and L. Van Gool, "Generative adversarial networks for extreme learned image compression," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 221–231.
[48] S. Iwai, T. Miyazaki, Y. Sugaya, and S. Omachi, "Fidelity-controllable extreme image compression with generative adversarial networks," in *Proc. 25th Int. Conf. Pattern Recognit.*, 2021, pp. 8235–8242.

[49] L. Wu, K. Huang, and H. Shen, "A GAN-based tunable image compression system," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, 2020, pp. 2323–2331.
[50] F. Mentzer, G. D. Toderici, M. Tschannen, and E. Agustsson, "High-fidelity generative image compression," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 11 913–11 924.
[51] Y. Zhang, K. Li, K. Li, B. Zhong, and Y. Fu, "Residual non-local attention networks for image restoration," in *Proc. Int. Conf. Learn. Representations*, 2019.
[52] M. Li, W. Zuo, S. Gu, J. You, and D. Zhang, "Learning content-weighted deep image compression," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 10, pp. 3446–3461, Oct. 2021.
[53] M. Li, W. Zuo, S. Gu, D. Zhao, and D. Zhang, "Learning convolutional networks for content-weighted image compression," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3214–3223.
[54] Y. Qian, X. Sun, M. Lin, Z. Tan, and R. Jin, "Entroformer: A transformer-based entropy model for learned image compression," in *Proc. Int. Conf. Learn. Representations*, 2022.
[55] Y. Zhu, Y. Yang, and T. Cohen, "Transformer-based transform coding," in *Proc. Int. Conf. Learn. Representations*, 2022.
[56] M. Lu, P. Guo, H. Shi, C. Cao, and Z. Ma, "Transformer-based image compression," in *Proc. Data Compression Conf.*, 2022, pp. 469–469.
[57] G. Toderici et al., "Variable rate image compression with recurrent neural networks," in *Proc. Int. Conf. Learn. Representations*, 2016.
[58] G. Toderici et al., "Full resolution image compression with recurrent neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 5306–5314.
[59] N. Johnston et al., "Improved lossy image compression with priming and spatially adaptive bit rates for recurrent networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4385–4393.
[60] X. Wang, K. Yu, C. Dong, and C. Change Loy, "Recovering realistic texture in image super-resolution by deep spatial feature transform," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 606–615.
[61] L. Ardizzone, J. Kruse, C. Rother, and U. Köthe, "Analyzing inverse problems with invertible neural networks," in *Proc. Int. Conf. Learn. Representations*, 2019.
[62] A. Lugmayr, M. Danelljan, L. V. Gool, and R. Timofte, "SRFlow: Learning the super-resolution space with normalizing flow," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 715–732.
[63] M. Xiao et al., "Invertible image rescaling," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 126–144.
[64] Y.-H. Ho, C.-C. Chan, W.-H. Peng, and H.-M. Hang, "End-to-end learned image compression with augmented normalizing flows," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2021, pp. 1931–1935.
[65] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2261–2269.
[66] J. Duda, "Asymmetric numeral systems: Entropy coding combining speed of Huffman coding with compression rate of arithmetic coding," 2013, *arXiv:1311.2540*.
[67] J. Liu, G. Lu, Z. Hu, and D. Xu, "A unified end-to-end framework for efficient deep image compression," 2020, *arXiv: 2002.03370*.
[68] T.-Y. Lin et al., "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 740–755.
[69] A. Paszke et al., "PyTorch: An imperative style, high-performance deep learning library," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 8026–8037.
[70] J. Bégaint, F. Racapé, S. Feltman, and A. Pushparaja, "CompressAI: A PyTorch library and evaluation platform for end-to-end compression research," 2020, *arXiv: 2011.03029*.
[71] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Representations*, 2015.
[72] D. Wang, W. Yang, Y. Hu, and J. Liu, "Neural data-dependent transform for learned image compression," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 17 379–17 388.
[73] G.-H. Wang, J. Li, B. Li, and Y. Lu, "EVC: Towards real-time neural image compression with mask decay," in *Proc. Int. Conf. Learn. Representations*, 2023.
[74] J. Guo, D. Xu, and G. Lu, "CBANet: Toward complexity and bitrate adaptive deep image compression using a single network," *IEEE Trans. Image Process.*, vol. 32, pp. 2049–2062, 2023.

**Shilv Cai** received the bachelor's degree from the College of Electrical and Information Engineering, Hunan University, Changsha, China, in 2018. He is currently working toward the PhD degree in Huazhong University of Science and Technology. He is interested in generative model and image compression.



**Liqun Chen** received the BS degree from the School of Materials Science and Engineering, Huazhong University of Science and Technology (HUST), Wuhan, China, and the PhD degree from the School of Artificial Intelligence and Automation, HUST, in 2012 and 2019, respectively. He is currently a lecturer with the School of Artificial Intelligence and Automation, HUST. His research interests include computer vision and image processing, in particular, embedded image processing system.

**Zhijun Zhang** received the bachelor's degree from College of Life Science and Technology, Huazhong University of Science and Technology, Wuhan, China, in 2015. She is currently working toward the PhD degree in Huazhong University of Science and Technology. She is interested in video analysis and action detection.

**Xiangyun Zhao** received the PhD degree from Northwestern University advised by Prof. Ying Wu. His research interest includes computer vision and machine learning. He actively publishes papers in computer vision conferences like CVPR/ICCV/ECCV. He did research intern with Microsoft Research Asia (Beijing, 2016), Adobe Research (San Jose, 2017), Baidu Research (Sunnyvale, 2018), NEC-Lab (San Jose, 2019) and Google Research (Seattle, 2020). He received top 10% paper award in ICIP 2015.

**Jiahuan Zhou** received the BE from Tsinghua University in 2013, the PhD degree from the Department of Electrical Engineering&Computer Science, Northwestern University in 2018. During 2018, he was a research intern with Microsoft Research, Redmond, Washington. From 2019 to 2022, he was a postdoctoral fellow and research assistant professor in Northwestern University. Currently, he is a Tenure-Track assistant professor with the Wangxuan Institute of Computer Technology, Peking University. His current research interests include computer vision, deep learning, and machine learning. He has authored 30 papers in international journals and conferences including *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *IEEE Transactions on Image Processing*, CVPR, ICCV, AAAI, ECCV, and so on. He serves as an area chair for CVPR, ICME, ICPR, an associate editor of *Springer Journal of Machine Vision and Applications* (MVA), a regular reviewer member for a number of journals and conferences, e.g., *IEEE Transactions on Pattern Analysis and Machine Intelligence*, IJCV, *IEEE Transactions on Image Processing*, CVPR, ICCV, ECCV, NeurIPS, ICML, and so on.

**Yuxin Peng** (Senior Member, IEEE) received the PhD degree in computer applied technology from Peking University, Beijing, China, in 2003. He is currently the Boya distinguished professor with the Wangxuan Institute of Computer Technology, Peking University. He has authored more than 200 papers, including more than 100 papers in the top-tier journals and conference proceedings. He has submitted 48 patent applications and been granted 39 of them. His current research interests mainly include cross-media analysis and reasoning, image and video recognition and understanding, and computer vision. He led his team to win the First Place in video semantic search evaluation of TRECVID ten times in the recent years. He won the First Prize of the Beijing Technological Invention Award in 2016 (ranking first) and the First Prize of the Scientific and Technological Progress Award of Chinese Institute of Electronics in 2020 (ranking first). He was a recipient of the National Science Fund for Distinguished Young Scholars of China in 2019, and the best paper award at MMM 2019 and NCIG 2018. He serves as the associate editor of *IEEE Transactions on Multimedia*, *IEEE Transactions on Circuits and Systems for Video Technology*, etc.

**Luxin Yan** (Member, IEEE) received the BS degree in electronic communication engineering, and the PhD degree in pattern recognition and intelligence system from the Huazhong University of Science and Technology (HUST) in 2001 and 2007, respectively. He is currently a professor with the School of Artificial Intelligence and Automation, HUST. His research interests include multi-spectral image processing, pattern recognition, and real-time embedded system.

**Sheng Zhong** received the PhD degree from the Huazhong University of Science and Technology, Wuhan, China, in 2005. He is a professor with School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan, China. His research interests include computer vision, pattern recognition, and intelligent system.

**Xu Zou** (Member, IEEE) received the PhD degree from Huazhong University of Science and Technology, Wuhan, China, in 2020. He is currently a lecturer with the School of Artificial Intelligence and Automation, Huazhong University of Science and Technology. His research interests include image processing, computer vision, and intelligent system.