Learning-based Compression for Noisy Images in the Wild

Pingping Zhang, Meng Wang, Baoliang Chen, Rongqun Lin, Xu Wang, *Member, IEEE,*
Shiqi Wang, *Senior Member, IEEE,* Sam Kwong, *Fellow, IEEE,*

***Abstract*—Digital images in real world applications typically undergo a wide variety of quality degradations before compression or re-compression. Existing learning based codecs are typically data-driven, relying on the predefined compression pipeline with pristine or high quality images as the input. However, the images in the wild may exhibit the substantially different characteristics compared to the high quality images, casting major challenges to the learning based image coding. In this paper, we propose a robust noisy image compression framework with the blind assumption on the specific noise type and level. The specifically designed encoder decomposes the representation of visual content into two types of features, including the Features that represent the Intrinsic Content (FIC) and the Features that account for Additive Degradation (FAD). As such, beyond the philosophy of faithfully reconstructing the given image with high fidelity, only FIC needs to be compactly represented and conveyed. The principled disentanglement strategy facilitates the removal of the redundancy from multiple perspectives (e.g., spatial, channel and content), ensuring the handling of a wide variety of noisy images in the wild. Extensive experimental results show that our model can achieve superior performance in terms of the ultimate quality and exhibit the strong generalizability across images degraded by a variety of means. The proposed scheme also points out a new research avenue on learning based compression for images in the wild, which is technically challenging but desirable in practice. Code: https://github.com/ppingzhang/NoisyIC.git**

***Index Terms*—End-to-end image compression, noisy images in the wild, generalization capability**

I. Introduction

End-to-end image compression has been making great progress benefiting from the data-driven deep neural networks in representing visual signals [1], [2]. However, before compression or re-compression, each stage in image acquisition and processing may introduce quality degradation. Existing end-to-end image codecs typically guarantee the fidelity between input and decoded images, while the gaps between pristine images and input images due to various types of corruption have been largely ignored. In fact, such corruption type and level are often unknown, and many causes may lead to quality degradations. Nevertheless, little work has been dedicated to compressing the images in the wild, and even less has been devoted to end-to-end compression of such images.

One major challenge in compressing the images in the wild lies in that the preservation of the high frequency information may involve undesirable from the input images. Typically, when optimizing the fidelity between the compressed and input images, noise is prone to be preserved as valuable information in high bit-rate compression. As a consequence, there is a typical phenomenon that the image quality cannot be improved with the increase on coding bits [3], [4]. Therefore, there is a consensus that the intrinsic content should be well preserved while the annoying artifacts are better to be removed. Various preprocessing techniques [5]–[8] attempt to restore the pristine images by removing the artifacts and enhancing the quality. However, such preprocessing heavily relies on the accurate identification of corruption patterns and effective preprocessing with diversified deep learning models. Moreover, the preprocessing is often independent of rate-distortion optimization (RDO) in end-to-end coding, while it may significantly influence the coding bits as well as the perceived quality.

From the above analyses, in this paper, we focus on the compression of the noisy images in the wild for the following reasons. First, during acquisition, the images may be contaminated by different types of noise due to certain constraints of lighting conditions, sensors and exposure conditions. Second, the compression of noisy image is more challenging, as the high frequency noise is typically preserved, degrading the final image quality and simultaneously increasing the coding bits. To compress such images, in particular the images with authentic noise, there are two desired properties of the codec, including the capability in reducing redundancy and the feasibility in representing the intrinsic content. In principle, both require an efficient transform, which motivates us to propose a learning-based compression model based on the principle of representing the input visual signals with the Features that represent the Intrinsic Content (FIC) and the Features that account for Additive Degradation (FAD). To this end, we focus on the end-to-end compression of noisy images in the wild and propose a universal and robust image compression framework, achieving the compact representation

Manuscript received December 16, 2021; revised July 12, 2022; accepted July 21, 2022. This work was supported in part by the National Natural Science Foundation of China under 62022002 and 61871270, in part by the Hong Kong Research Grants Council, General Research Fund (GRF) under Grant 11203220, in part by the Hong Kong Innovation and Technology Commission (InnoHK Project CIMDA). Corresponding author: Shiqi Wang (email:shiqwang@cityu.edu.hk).

Pingping Zhang, Meng Wang, Baoliang Chen, Rongqun Lin, Shiqi Wang and Sam Kwong are with the Department of Computer Science, City University of Hong Kong, Hong Kong, China, (email: ppingyes@gmail.com; mwang98-c@my.cityu.edu.hk; blchen6-c@my.cityu.edu.hk; rqlin3-c@my.cityu.edu.hk; shiqwang@cityu.edu.hk; cssamk@cityu.edu.hk).

Xu Wang is with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518060, China, and also with Guangdong Laboratory of Artificial Intelligence and Digital Economy (SZ), Shenzhen University, Shenzhen 518060, China, (email: wangxu@szu.edu.cn).

of features and disentangling FIC and FAD from coarse to fine. Meanwhile, our model is featured with low encoding and decoding complexity. Moreover, the decoding speed is faster than the encoding speed due to the adoption of an asymmetric encode-decode architecture, which significantly reduces network parameters and accelerates the inference speed, revealing potential benefits in real-world applications. Overall, the main contributions of this paper are summarized as follows,

- We propose a learning based compression scheme for noisy images in the wild, based on the principle of the redundancy removal and efficient intrinsic visual content representation.
- We propose to disentangle the visual representation into the FIC and FAD features, where the FIC features account for the content to be preserved and the FAD features need to be effectively removed.
- Extensive experiments and analyses verify the effectiveness of our model, which delivers better rate-distortion (RD) performance on different types of noisy images in the wild compared to the state-of-the-art learning based compression methods.

II. Related Works

A. Learning based Image Compression

Image compression targets at compactly representing image signals to facilitate transmission and storage. Numerous image compression standards have been developed in the past decades, including the JPEG [9], the JPEG2000 [10], the High Efficiency Video Coding (HEVC)/H.265 [11], and the Versatile Video Coding (VVC)/H.266 [12]. The traditional image coding paradigm is employed delicately with prediction, transform coding and entropy coding modules to eliminate redundancies existing in image data, and accurate entropy estimation benefits the optimization of codecs, such that the overall framework is optimized with the RDO.

In contrast with the traditional transform, e.g., discrete cosine transform (DCT), discrete Fourier transform (DFT) and discrete wavelet transform (DWT), the learning-based methods facilitate the compact representation of visual signals in a data driven manner. Benefiting from the capacity of deep learning models, recent years have witnessed the tremendous development of deep-learning based transform coding. These researches have revealed that neural networks are capable of nonlinear modelling of visual signals. Ballé *et al.* [1] proposed a nonlinear transform-based end-to-end image compression framework with generalized divisive normalization (GDN) to model image content, which shows an impressive capacity for image compression. Subsequently, several end-to-end image compression algorithms have been proposed by transforming the input into a latent code. A convolutional neural network (CNN) model is designed as a deep learning-based transform [13] to achieve better decorrelation and energy compaction. In addition, attention modules [14] are introduced to strengthen the transform capabilities of the compression algorithm in order to obtain higher compression performance.

Entropy estimation serves as an important step in learning-based image compression. The commonly-used factorized entropy model [1] is based on the hypothesis that the individual latent representation is independent, though this condition is difficult to be guaranteed. The hyperprior network [15] has been proposed to extract side information from the latent representation, which can represent latent distributions and enhance the latent coding entropy estimation, improving overall coding performance. Inspired by the success of autoregressive priors in probabilistic generative models, Minnen *et al.* [16] proposed to combine the spatial context model with a hyperprior for conditional entropy estimation in order to improve conditional entropy estimation. The context model is used to predict the likelihood of unknown codes based on previously decoded latent representation. However, even when processing relatively small images, it is evident that a serial context model is time-consuming in encoding and decoding. To address the above problem, Hu *et al.* [17] proposed a coarse-to-fine entropy model to reduce redundancy in the latent representation, and it enjoys much faster decoding speed than the context based model.

B. Corrupted Image Compression

Due to the absence of pristine images in real-world applications, efforts have been devoted to distorted image/video compression [3], [4], [18]. The conventional solution is to preprocess before compression, e.g., filtering the noise. As a result, these algorithms consider preprocessing and compression as separate operations but ignore the advantage of joint optimization. Chen *et al.* [4] built the relationship of the lower bound quantization parameter (QP) and the noise variance. In this manner, it can simultaneously perform rate control and video denoising with a lower bound QP constraint. Subsequently, Li *et al.* [3] investigated the properties of rate-distortion performance and proposed a pre-analytical model based on deep learning to denoise images with the end-to-end compression framework. Moreover, they proposed a new data-driven technique for defending noisy input without previous knowledge of the noise level. However, they did not provide a solution for compressing distorted images from distinct domains. The modeling of the images in the wild, as well as the quality assessment, are still very challenging tasks [19]. JPEG AI targets to develop a learning-based image coding standard, which achieves compact representation for human viewing and support a wide range of applications. Moreover, the JPEG-AI call for proposals also take the compressed-domain denoising into consideration as a task [20].

C. Domain Generalization and Contrastive Learning

Domain generalization has received increasing attention in recent years, and many solutions have proposed, e.g., domain alignment [21], data augmentation [22], learning disentangled representations [23], [24]. An intuitive way to achieve disentangled representation learning is to decompose a model into two parts: domain-specific and domain-agnostic. Based on SVMs, Khosla *et al.* [25] decomposed a classifier into domain-specific biases and domain-agnostic weights, and only

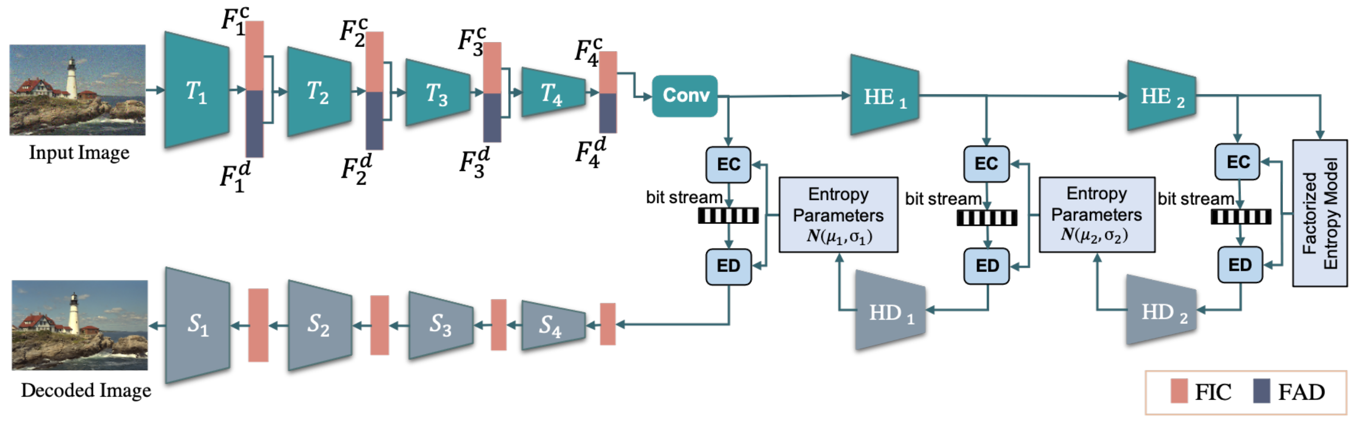


Fig. 1. Illustration of the encoding and decoding framework. In the encoder, given an input image, it is first transformed into the representation of FIC and FAD. Only FIC needs to be compactly represented and conveyed. In the decoding phase, after the bitstreams are decoded, the compact features are projected to the standard image representation via four synthetic modules.

maintained the latter one when dealing with unseen domains. This approach was later extended to neural networks in [23]. Moreover, one can also design domain-specific modules such as [26] where domain-specific binary masks are imposed on the final feature vector to distinguish between domain-specific and domain-invariant components.

Obviously, learning domain invariant representations is crucial in domain generalization. Numerous types of distortion may occur in the image in the wild. Many researchers investigate the invariant intrinsic properties of the image to reconstruct clean images. Li *et al.* [23] proposed a conditional invariant adversarial network which can guarantee the domain-invariance property. Du *et al.* [27] proposed an adversarial domain adaptation approach to develop robust representations under feature and image domain restrictions for image restoration. Inspired by the feature disentanglement, our work adopts this design philosophy that input images are decomposed into the FIC and FAD, where FIC is the domain-invariant component and FAD is the domain-specific component. Therefore, only FIC is compactly represented and conveyed for the final reconstruction in the decoder.

Contrastive learning [28] learns representations by distinguishing positive and negative examples. Moreover, contrastive learning processes the data in feature spaces to facilitate the model optimization, leading to robust generalization capability. The training objectives have been widely studied in contrastive learning to improve the generalization capability. Contrastive loss [29] is one of the earliest training objectives employed as the learning metric in a contrastive fashion. Triplet loss [30] was originally proposed to learn face recognition models of the same identity with different poses and angles, and now it is a prevalent training objective applied in various applications. Our work adopts the triplet loss as the optimization objective to efficiently disentangle the FIC and FAD.

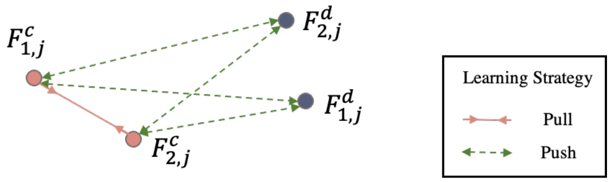


Fig. 2. Illustration of the learning strategy for disentanglement of FIC and FAD.

III. Learning Based Compression Scheme for Noisy Images in the Wild

A. Overview of the Compression Framework

The overall architecture of the proposed compression scheme is illustrated in Fig. 1. Hence, in the encoder, given an input image, it is first decomposed into the representations including FIC and FAD. Herein, FIC ($F_j^c, j \in \{1, 2, 3, 4\}$) equips the ability to characterize intrinsic visual information. On the contrary, FAD ($F_j^d, j \in \{1, 2, 3, 4\}$) refers to additive degradation such as noise. To reliably disentangle features, FIC is sequentially represented across four transform modules from T_1 to T_4 with the assistance of the FAD, supplying more flexibility and incorporating more variations than the single stage method. Ultimately, only FIC is compactly represented and efficiently compressed via the coarse-to-fine entropy model. In the decoding phase, after the bitstreams are decoded, the compact features are projected to the standard image representation via four synthetic modules ($S_j, j \in \{1, 2, 3, 4\}$).

The proposed decomposition mechanism is rooted in the widely accepted view that only the intrinsic visual information that governs the visual perception and understanding needs to be conveyed. Hence, to learn the disentangled FIC and FAD, we form the training triplets each of which consists three images, including a ground-truth image (I_p) as well as two corrupted images (I_1 and I_2) from the same ground-truth image with distinct distortion types. An effective strategy to disentangle FIC and FAD is to pull the FIC closer, and push away FIC against FAD, as illustrated in Fig. 2. To this end, we first adopt four transform layers (denoted as

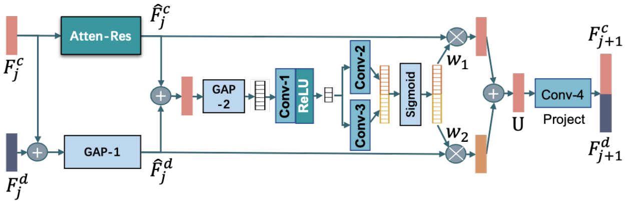
T_1 to T_4 in Fig. 1) to decompose the $F^c_{i,j}$ (FIC) and $F^d_{i,j}$ (FAD, $i \in \{1,2\}$, $j \in \{1,2,3,4\}$) from image I_i and the design detail of our transform module will be described in Sec.III.B. Subsequently, the triplet loss [30] is performed on $F^c_{i,j}$ and $F^d_{i,j}$, aiming to minimize the distance between $F^c_{1,j}$ and $F^c_{2,j}$ while maximize the distance between $F^c_{k,j}$ and $F^d_{l,j}$ ($k, l \in \{1,2\}$ and $k \neq l$). In this manner, FIC and FAD can be decomposed progressively, and only compact FIC is compressed and conveyed to the decoder. Moreover, the design philosophy of the decoder follows the principle of lightweight and high efficiency, which employs a lightweight ResNet [31] as the inverse transform module to reconstruct the images.

B. Multi-scale Disentanglement Encoder

The proposed multi-scale disentanglement encoder is capable of translating the images into the latent code characterized by FIC. The advantages that account for the design philosophy of the multi-scale disentanglement encoder are two-fold. On the one hand, such a multi-scale decomposition strategy equips the capability of gradually disentangling FIC from coarse to fine. On the other hand, the multi-scale disentanglement is efficient in characterizing the visual details at different resolutions, thereby removing content, channel, and spatial redundancies. Herein, the transform module is the main component in the encoder.

The primary goal of the transform module is to remove redundancy, project the features into a separable space and extract the clean FIC. Our proposed transform module minimizes feature redundancy, efficiently disentangles features, and achieves a good compromise between rate and distortion. More specifically, the transform module first maintains the FIC while exploring more intrinsic contents. Then, the selective feature structure is adopted to adjust the relationship of channels to minimize channel redundancy. Finally, the features are projected to a compact and separable space. The detailed transform module is shown in Fig. 3. To extract the clean FIC, F^c_j is first processed via the residual attention module (Atten-Res) [32], as shown in Fig. 4. Through the residual structure in the Atten-Res module, the learned residual masks can enhance FIC while suppressing the additive degradation. Meanwhile, to explore more intrinsic contents, F^c_j and F^d_j with the dimension of $C \times H \times W$ are combined together via the element-wise sum operation, where H and W represent the height and width of the features, respectively, and C depicts the channel number of features. Then, the global average pooling operation is utilized to filter the noise. In this way, the informative cues can be explored, such as the certain portion of the intrinsic content, which can be mixed with FAD. In this way, the informative cues can be explored again, such as part of intrinsic contents, which are mixed with FAD and not disentangled in the previous transform.

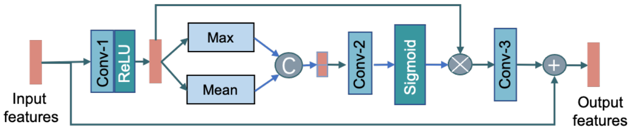
To reduce the channel correlation, we need to collect information from each channel. As a result, an element-wise sum operation is used to merge two parallel feature streams ($\hat{F}^c_j$ and $\hat{F}^d_j$) as shown in Fig. 3. The weights of features are then adjusted to obtain the compact representation in the channel dimension. Inspired by [33], we adopt the selective



Layer	Input Dimension	Output Dimension	Kernel Size
Atten-Res	$C{\times}H{\times}W$	$C{\times}H{\times}W$	–
GAP-1	$C{\times}H{\times}W$	$C{\times}H{\times}W$	5×5
GAP-2	$C{\times}H{\times}W$	$C{\times}1{\times}1$	$H{\times}W$
Conv-1	$C{\times}1{\times}1$	$C/2{\times}1{\times}1$	1×1
Conv-2	$C/2{\times}1{\times}1$	$C{\times}1{\times}1$	1×1
Conv-3	$C/2{\times}1{\times}1$	$C{\times}1{\times}1$	1×1
Conv-4	$C{\times}H{\times}W$	$2C{\times}H/2{\times}W/2$	3×3

Feature	Dimension
F^c_j, F^d_j	$C{\times}H{\times}W$
$\hat{F}^c_j, \hat{F}^d_j$	$C{\times}H{\times}W$
w_1, w_2	$C{\times}H{\times}W$
U	$C{\times}H{\times}W$
F^c_{j+1}, F^d_{j+1}	$C{\times}H/2{\times}W/2$

Fig. 3. The architecture of the transform module. The left table shows the parameters of different layers, and the right table gives the detailed dimension information of features.



Layer	Input Dimension	Output Dimension	Kernel Size
Conv-1	$C{\times}H{\times}W$	$C{\times}H{\times}W$	3×3
Max	$C{\times}H{\times}W$	$C{\times}1{\times}1$	–
Mean	$C{\times}H{\times}W$	$C{\times}1{\times}1$	–
Conv-2	$C{\times}H{\times}W$	$C{\times}H{\times}W$	3×3
Conv-3	$C{\times}H{\times}W$	$C{\times}H{\times}W$	3×3

Fig. 4. The architecture of the Atten-Res Module. The table provides the parameters of layers.

feature structure to adjust features and reduce the channel correlation. Specifically, two features ($\hat{F}^c_j$ and $\hat{F}^d_j$) are fused to yield global feature descriptors with a global average pooling operation over the spatial dimension to measure channel-wise statistics. Subsequently, we employ a channel-downscaling convolution layer to construct the compact feature representation. Afterward, the feature vector passes through two parallel channel-upscaling convolution layers and provides two feature descriptors. The selective operator applies the Softmax function to generate attention activations w_1 and w_2, such that we can adaptively adjust the feature for a more compact representation from the channel dimension. Herein, the whole feature adjustment and aggregation procedure is defined as $\mathbf{U} = w_1 \cdot \hat{F}^c_j + w_2 \cdot \hat{F}^d_j$.

Following this, we project the features to a compact and separable space via the convolution with a stride of 2. As such, FIC and FAD can be extracted from different channels, where we equally divide the features into FIC ($F^c_{j+1} \in \mathbb{R}^{C \times H/2 \times W/2}$) and FAD ($F^d_{j+1} \in \mathbb{R}^{C \times H/2 \times W/2}$).

The transform module decomposes FIC and FAD at a lower spatial resolution from the previous layer. After four coarse-to-fine transform operations, we can finally obtain the clean and compact FIC.

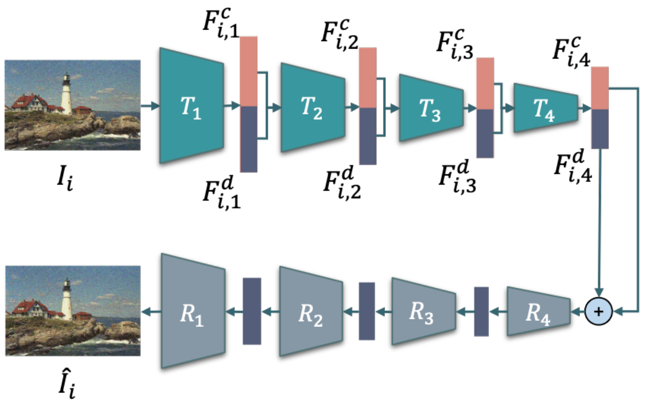


Fig. 5. Reconstruction of input images. We extract the compact representation $F^c_{i,4}$ and $F^d_{i,4}$ after four transformations, and add the $F^c_{i,4}$ and $F^d_{i,4}$ together to obtain the compact representation of the input I_i. Through four reconstruction modules from R_4 to R_1, the image can be reconstructed as $\hat{I}_i$.

C. Entropy Model

A fast and accurate entropy estimation model is critical to the efficiency of the codec. Inspired by the coarse-to-fine entropy model [17], we utilize a multi-layer conditioning framework to estimate the probability of each symbol, which can be described as follows,

$$P_{\mathbf{X}}(\mathbf{X}) = P_{\mathbf{Z}}(\mathbf{Z}) P_{\mathbf{Y}|\mathbf{Z}}(\mathbf{Y} \mid \mathbf{Z}) P_{\mathbf{X}|\mathbf{Y}}(\mathbf{X} \mid \mathbf{Y}), \tag{1}$$

where $\mathbf{X}$ is the latent representation after transformation, and $\mathbf{Y}$ denotes a hyper representation extracted via the first hyperprior encoding module ($\mathbf{HE}_1$). Herein, $\mathbf{Y}$ involves more information to provide accurate conditional modelling. As a result, an extra hyperprior coding module ($\mathbf{HE}_2$) is introduced to extract a higher-level representation $\mathbf{Z}$.

Assuming that the conditional distribution of each element in $\mathbf{X}$ and $\mathbf{Y}$ follows the Gaussian distribution, the probability estimation network predicts the mean and scale of the Gaussian distribution, as shown in Fig. 1. More specifically, $\mathbf{HE}_1$ contains three convolution layers appended with leaky ReLU [34] layers except for the last convolution layer. Analogously, $\mathbf{HE}_2$ consists of three convolution layers appended with ReLU layers except for the last convolution layer. Regarding the probability estimation of $\mathbf{Z}$, the coarse-to-fine model [17] assumes that the $\mathbf{Z}$ obeys the zero-mean Gaussian distribution, of which the variance is a trainable parameter. However, different images may possess distinct characteristic. As such, employing a fixed distribution to model $\mathbf{Z}$ is insufficient for representation. By contrast, the factorized entropy model can well fit to arbitrary densities [1]. Thus, we employ the factorized model instead of the zero-mean Gaussian model to more precisely estimate the probability. Finally, the rate model can be expressed as follows,

$$\begin{aligned} R = \mathbb{E}_{\mathbf{X}|\mathbf{Y}}[-\log(P_{\mathbf{X}|\mathbf{Y}})] + \\ \mathbb{E}_{\mathbf{Y}|\mathbf{Z}}[-\log(P_{\mathbf{Y}|\mathbf{Z}})] + \mathbb{E}_{\mathbf{Z}}[-\log(P_{\mathbf{Z}})]. \end{aligned} \tag{2}$$

In particular, the hyperprior decoding modules ($\mathbf{HD}_1$ and $\mathbf{HD}_2$) consist of two deconvolution layers with leaky ReLU and ReLU operations, respectively. The final layer of $\mathbf{HD}_1$ and $\mathbf{HD}_2$ is the convolution layer, such that the hyperprior decoder shares a symmetric structure with respective to the hyperprior encoder.

D. Loss Functions

In summary, the rate-distortion (RD) loss function ($\mathcal{L}_{RD}$) in our proposed method includes the feature disentanglement loss $\mathcal{L}_{DIS}$, the content reconstruction loss $\mathcal{L}_{CR}$ and the bitrate (R) for the image encoding, which is given by:

$$\mathcal{L}_{RD} = \mathcal{L}_{DIS} + \mathcal{L}_{CR} + R. \tag{3}$$

Regarding the $\mathcal{L}_{DIS}$, two input images I_1 and I_2 are utilized for the FIC and FAD extraction and disentanglement. In particular, the I_1 and I_2 share the same content while they are corrupted by different noise types. The triplet loss $\mathcal{L}_{trp}$, which has been widely adopted in contrastive learning, is employed to disentangle FIC and FAD of each image. As shown in Fig. 2, the $\mathcal{L}_{trp}$ is given by,

$$\begin{aligned} \mathcal{L}_{trp} = \frac{1}{4}\sum_{j=1}^{4}\left[\left\|F^c_{1,j} - F^c_{2,j}\right\|_1 - \left\|F^c_{1,j} - F^d_{2,j}\right\|_1 + \alpha\right]_+ \\ + \frac{1}{4}\sum_{j=1}^{4}\left[\left\|F^c_{2,j} - F^c_{1,j}\right\|_1 - \left\|F^c_{2,j} - F^d_{1,j}\right\|_1 + \alpha\right]_+, \end{aligned} \tag{4}$$

where α is a preset margin enforced between positive and negative pairs. $\|\cdot\|_1$ represents the $\mathcal{L}_1$-norm. $F^c_{i,j}$ and $F^d_{i,j}$ ($i \in \{1,2\}, j \in \{1,2,3,4\}$) are the FIC and FAD decomposed via j-th transform module from the i-th input image. With $\mathcal{L}_{trp}$, the distance among FIC ($F^c_{i,j}$) is minimized, while the distances between $F^c_{1,j}$ and $F^d_{2,j}$ and between $F^c_{2,j}$ and $F^d_{1,j}$ are maximized. To ensure the full content information are extracted, the noisy image reconstruction loss $\mathcal{L}_{nir}$ is further utilized. More specifically, as shown in Fig. 5, for the i-th input noisy image, the extracted $F^c_{i,4}$ and $F^d_{i,4}$ are first added and treated as the input of the reconstruction module (denoted as R_4 to R_1 in Fig. 5). Then the reconstructed noisy image $\hat{I}_i$ can be acquired by the pixel-wise loss defined as follows,

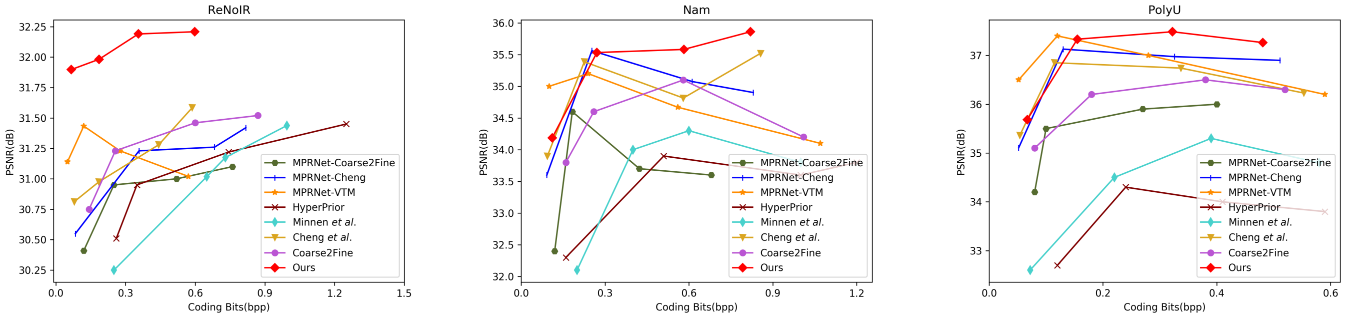
$$\mathcal{L}_{nir} = \frac{1}{N}\left\|\hat{I}_1 - I_1\right\|_2^2 + \frac{1}{N}\left\|\hat{I}_2 - I_2\right\|_2^2, \tag{5}$$

where N is the number of pixels of each image and $\|\cdot\|_2$ represents the $\mathcal{L}_2$-norm. As such, the $\mathcal{L}_{DIS}$ consisting of $\mathcal{L}_{trp}$ and $\mathcal{L}_{nir}$ is given by,

$$\mathcal{L}_{DIS} = \lambda_{trp}\mathcal{L}_{trp} + \lambda_{nir}\mathcal{L}_{nir}, \tag{6}$$

where λ_{trp} and λ_{nir} are two parameters to adjust the weights of the triplet loss and the noisy image reconstruction loss, respectively.

Regarding the content reconstruction loss ($\mathcal{L}_{CR}$), we aim to reconstruct the content information from the encoded FIC. In particular, the ground-truth image (denoted as I_p) is introduced and treated as the target image for the reconstruction quality evaluation. Herein, instead of only using the widely used Mean Squared Error (MSE) loss (denoted as $\mathcal{L}_{mse}$) for supervision,



(a) ReNoIR (b) Nam (c) PolyU

Fig. 6. Rate-distortion performance on the noisy images in the wild. The PSNR is obtained by the comparisons between ground-truth images and the decoded images. The coding bits and PSNR are obtained by the average of all test images. The training dataset contains BSD500 with AWGN and SIDD.

the perceptual loss [35] (denoted as $\mathcal{L}_{pc}$) is further adopted for visually pleasant reconstruction, which is defined as follows,

$$\mathcal{L}_{pc}=\sum_{i=1}^{2}\sum_{k=1}^{5}\left\|\phi^{k}\left(I_{i}^{c}\right)-\phi^{k}\left(I_{p}\right)\right\|_{2}^{2}, \tag{7}$$

where ϕ^k means the k-th feature extractor in VGG19 [36], and i is the index of the decoded image. As such, our $\mathcal{L}_{CR}$ is a combination of the $\mathcal{L}_{mse}$ and $\mathcal{L}_{pc}$ which is defined as follows,

$$\mathcal{L}_{CR}=\lambda_{mse}(\mathcal{L}_{mse}(I_1^c,I_p)+\mathcal{L}_{mse}(I_2^c,I_p))+\lambda_{pc}\mathcal{L}_{pc}, \tag{8}$$

where I_1^c and I_2^c are the decoded images from I_1 and I_2, respectively. λ_{mse} and λ_{pc} are two hyper-parameters. In addition, as two corrupted images (I_1 and I_2) are involved in the training phase, the bitrate constraint R in Eqn. (3) consists two terms *i.e.,* R_1 and R_2, which represent the bitrates consumed by I_1 and I_2, respectively. Finally, the total RD loss defined in Eqn. (3) can be rewritten as follows,

$$\begin{aligned}\mathcal{L}_{RD}&=\mathcal{L}_{DIS}+\mathcal{L}_{CR}+R\\&=\lambda_{trp}\mathcal{L}_{trp}+\lambda_{nir}\mathcal{L}_{nir}+\lambda_{mse}(\mathcal{L}_{mse}(I_1^c,I_p)\\&+\mathcal{L}_{mse}(I_2^c,I_p))+\lambda_{pc}\mathcal{L}_{pc}+R_1+R_2.\end{aligned} \tag{9}$$

The whole network is trained in an end-to-end manner, ensuring the overall optimized performance.

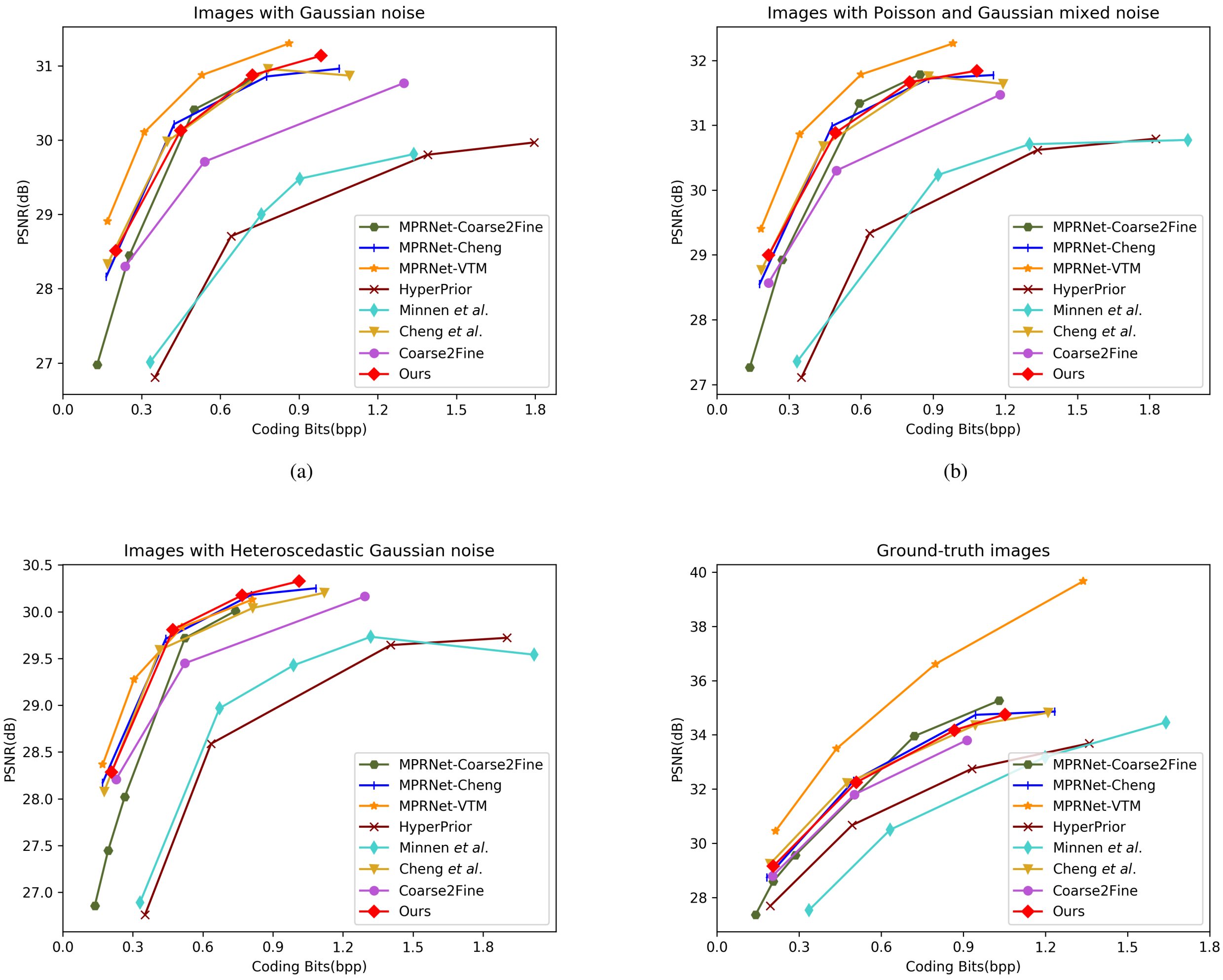
IV. Experimental Results

A. Implementation Details

1) Training and Testing Data: Regarding the training data, the ground-truth and corresponding corrupted images are formed as the triplet pairs. First, we adopt the images from Flicker [37] and Berkeley Segmentation Data Set 500 (BSD500) [38] as ground-truth images, which are subsequently degraded by additive white Gaussian noise (AWGN) with different levels (σ={1, 10, 20, 30, 40, 50}) for generating synthetic noise images. Moreover, the Smartphone Image Denoising Dataset (SIDD) [39] with the real noise images and corresponding ground-truth images is also involved. In particular, regarding SIDD, the ground-truth images as well as the acquired noisy images captured by different cameras in different lighting conditions form the triplet pairs. To prevent over-fitting in the training phase, we also augment the datasets via randomly flipping the images horizontally and vertically.

In the testing stage, we are particularly interested in the compression performance of noisy images in the wild which are authentically distorted. The real-world datasets, including Nam [40], ReNoIR [41] and PolyU [42] are involved for testing. In general, real-world noise images captured by different devices are featured with multiple distortion levels and different types of noise. The Nam [40] dataset includes images which are captured by three cameras containing 11 static scenes. Each scene involves 500 JPEG images. For evaluation, we employ the officially released sub-dataset of Nam for testing, which contains 15 images with the resolution 512×512 patches. In addition, we include the ReNoIR dataset in testing, which contains noisy-clean pairs. The noisy images are obtained naturally from short-time exposure in low-light scenes, and the clean counterparts are obtained from long-time exposure of the same scene. Again, we center crop the images to 1024×1024 patches and randomly select 20 pairs for evaluation. The PolyU [42] dataset is also considered in our experiment. The images in the PolyU dataset are captured via five different cameras in 40 scenes, forming the noisy-pristine pairs. The PolyU dataset provides a small bunch dataset which is composed with 512×512 patches extracted from the high-resolution images. We employ such small bunch dataset in our experiment. In addition to these real-world noisy images, we also attempt to compress the images with synthetic noise. In particular, the images with synthetic noise are generated with the Kodak dataset [43] and the AWGN (standard deviations 15, 25 and 45). It is worth mentioning that the noise levels characterized by the standard deviation in training and test datasets are different. In addition, we employ the complex noise models, *e.g.*, heteroscedastic Gaussian [44] $n_i \sim \mathcal{N}\left(0, \alpha^2 x_i + \delta^2\right)$ with $\alpha = 40$ and $\delta = 10$, and Gaussian-Poisson [45], to generate the images with different types of synthetic noise.

2) Training and Testing Settings: Regarding the experimental settings, each training batch including eight patches is extracted as inputs with the size of 256×256. Regarding the settings of λ_{trp}, λ_{nir}, λ_{mse} and λ_{pc}, λ_{mse} is set to $n \times 255$ wherein $n \in${1, 5, 20, 50}) corresponds to different



(a) (b) (c) (d)

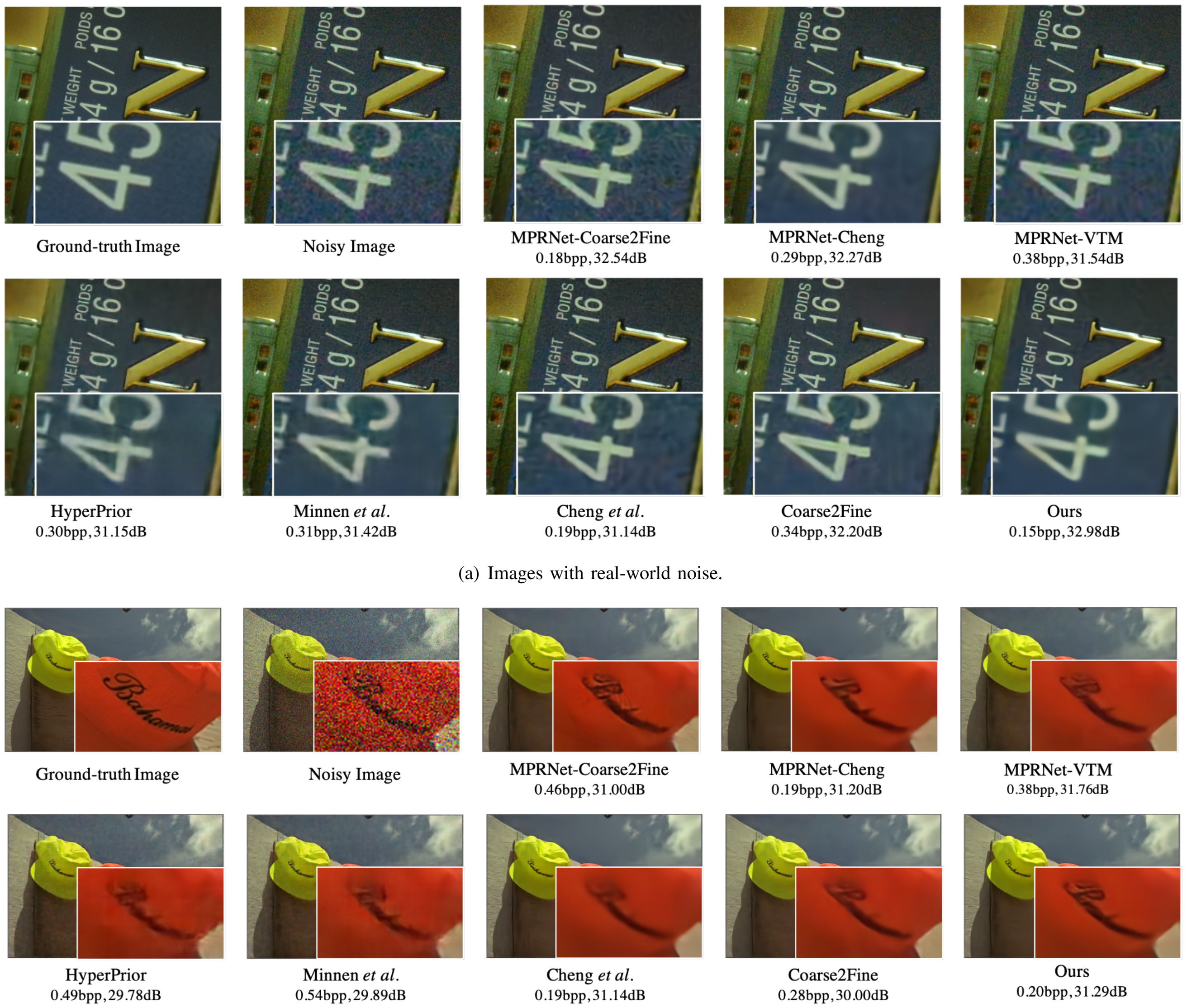
Fig. 7. The rate-distortion performance for the synthetic noise images and ground-truth images. (a) Compression performance of images with Gaussian noise. (b) Compression performance of images with Poisson and Gaussian noise. (c) Compression performance of images with Heteroscedastic Gaussian noise. (d) Compression performance of ground-truth images. Again, the PSNR values are obtained based on the comparisons with the ground-truth images and the decoded images.

bitrate points. Then, the parameters are set as: $\lambda_{nir} = \lambda_{mse}$, $\lambda_{trp} = 0.1 \times \lambda_{mse}$, and $\lambda_{pc} = 0.5 \times \lambda_{mse}$. Adam [46] is used as the optimizer with default parameters, wherein the learning rate is initialized as 1e-4. The network is implemented in the PyTorch framework and trained with an NVIDIA Tesla V100 GPU.

In the testing phase, we choose five state-of-the-art compression models for comparison, including the VVC test model (VTM) [47], Cheng *et al.*'s [14], Coarse2Fine [17], HyperPrior [15] and Minnen *et al.*'s [16]. Existing learning-based codecs are trained with pristine images. For fair compression, we retrain learning-based codecs, e.g., Cheng *et al.*'s, Coarse2Fine, HyperPrior and Minnen *et al.*'s, with our training dataset. To comprehensively evaluate the performance of the proposed method, the preprocessing then compression paradigm is involved for comparison. More specifically, preprocessing is firstly conducted through a denoising model MPRNet [48]. Subsequently, the denoised images are compressed with the VTM (denoted as MPRNet-VTM) and the learning-based codecs, including Cheng *et al.* (MPRNet-Cheng) and Coarse2Fine (MPRNet-Coarse2Fine) network. For other models without preprocessing, the training and testing procedures are conducted under the same conditions. We provide the rate-distortion performance, where the distortion is obtained by comparing the decoded image against the ground-truth image.

B. *Performance Comparisons*

1) Real-World Noisy Image Compression: The rate-distortion performance of different datasets is shown in Fig. 6. Typically, there are several observations. First, regarding the results on the ReNoIR dataset, most methods perform poorly due to the large gap of noise types between the training and testing data, resulting in unsatisfactory capabilities on the compression of images with unseen corruptions. Second, a high-quality peak is observed on the Nam dataset, beyond



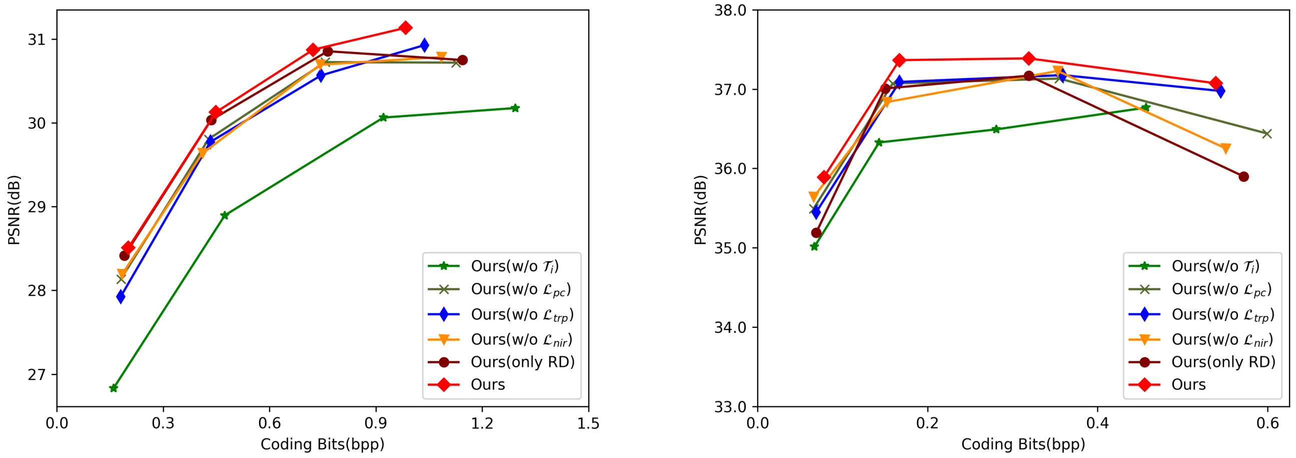
(a) Images with real-world noise.

(b) Images with Gaussian noise (σ=45)

Fig. 8. The visualization of decoded images with different methods. (a) The ground-truth and noisy images are from the Nam [40] dataset. (b) The ground-truth image is from the Kodak [43] dataset, and the noisy image is corrupted by Gaussian noise with σ=45.

which the increase of bitrate cannot be rewarded with better coding quality. More specifically, MPRNet-Cheng performs well at low bitrates. However, the rate-distortion performance drops significantly with the increase of bitrates as more noise is preserved. Third, preprocessing based codecs, such as MPRNet-VTM and MPRNet-Coarse2Fine, cannot deliver the promising performance. We have also shown the decoded images with different methods in Fig. 8 (a). It is apparent that the methods with preprocessing such as MPRNet-Coarse2Fine, MPRNet-Cheng and MPRNet-VTM, may lead to the failure since the poor generalization ability in the preprocessing. By contrast, our model can produce relatively satisfactory quality when compared with other codecs, e.g., Minnen *et al.*'s and Cheng *et al.*'s under a similar bitrate level. The advantage of our model compared with the "preprocessing-VTM" scheme lies in that our model can process many kinds of noisy images, no matter whether they are seen or unseen in training. Our model shows the reasonable generalization capability compared with the "preprocessing-VTM" scheme since the prior knowledge on the noise type is better to be incorporated into "preprocessing-VTM".

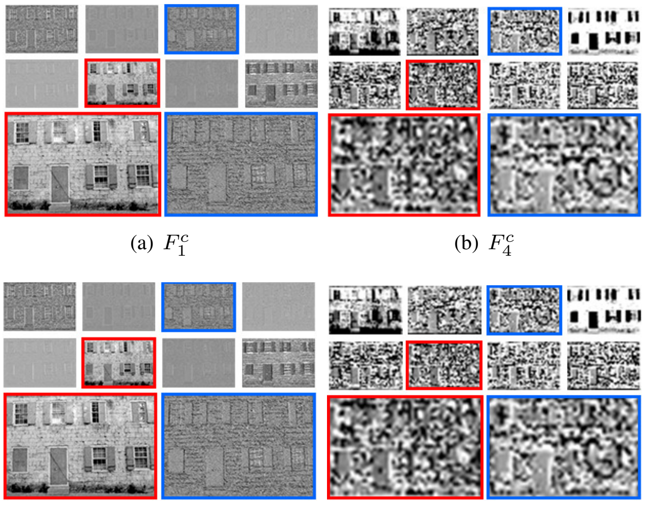
2) Compression of Images with Synthetic Noise: Furthermore, we evaluate our model on the images with synthetic noise. The testing dataset with Gaussian-Poisson noise and Heteroscedastic Gaussian noise share certain similarities to the training dataset with Gaussian noise. The rate-distortion performance is illustrated in Fig. 7. Regarding the synthetic Gaussian noise datasets which contain identical type of noise with the training dataset, as the state-of-the-art denoising method MPRNet can process the known noise type efficiently especially at low noise levels, high compression efficiency is achieved such as MPRNet-VTM. Moreover, it is interesting to observe that most learning-based codecs cannot achieve satisfactory performance in high bitrate coding scenario, whereas



(a) Gaussian Noise

(b) PolyU real noise dataset

Fig. 9. The rate-distortion performance comparisons in our ablation studies.



(a) F_1^c

(b) F_4^c

(c) F_1^p

(d) F_4^p

Fig. 10. Visualization of the sampled feature maps of F_1^c, F_4^c, F_1^p and F_4^p. The first and second row are the feature maps extracted from the noisy image and pristine image, respectively. The 3rd and 6th features are enlarged for better visualization, and borders of the same color are the same feature map with different sizes.

the proposed model could perform well, exhibiting stable increases regarding the quality of the reconstructed images. Similar trends could be observed on the Poisson and Gaussian mixed noise datasets, as shown in Fig. 7 (b). In addition, when encountering a new synthetic noise type such as the Heteroscedastic Gaussian noise that is totally unseen in the training data, the proposed model successfully surpasses the VTM at high coding bitrate. Meanwhile, comparing with the codecs without preprocessing, which may lack the capability of disentangling the additive noise, the proposed method consistently delivers promising results in both the low bitrate and high bitrate coding scenarios. Regarding quality of the decoded images, we visualize exemplified images degraded by Gaussian noise with σ=45, which are compressed with different models, as shown in the first row of Fig. 8 (b). MPRNet-VTM achieves better PSNR results, whereas the texture details may be distorted such as the details of the initial letter, due to the excessive smoothness in the preprocessing stage. The compression without preprocessing schemes could reserve more fine details, and meanwhile preserve the noise. As such, the proposed method strikes an excellent balance in detail-preserving and noisy image compression, yielding reconstructions with satisfactory quality.

3) Noise-free Image Compression: To evaluate the generalization capability of our model, we adopt the pristine Kodak dataset for testing, and the results are shown in Fig. 7 (d). It can be observed that the VTM reveals remarkable compression performance gain on the pristine images. The proposed model surpasses the learning-based models, such as the Minnen *et al.*'s [16], Cheng *et al.*'s [14], Coarse2Fine [17], and HyperPrior [15]. Moreover, preprocessing based compression methods perform well on these noise-free images as well. This is explainable as the associated compression models are specifically trained with preprocessed images, such that the noise-free images could be effectively compressed. Nevertheless, our proposed model achieves the close performance at the low bitrate compression and outperforms other learning-based codecs without preprocessing.

C. Ablation Studies

To evaluate the efficiency of each component of our codec, we conduct the ablation studies on the transform module, the entropy model, and the objective function, respectively.

1) Transform Module: The designed transform module aims for feature disentanglement and redundancy removal. Herein, to verify its functionality, we ablate the transform modules T_i (in Fig. 1) and replace them with the trivial convolution layers. The ablation results are shown in Fig. 9 and we denote it as **Ours (w/o $\boldsymbol{T}_\mathrm{i}$)**. Compared with our

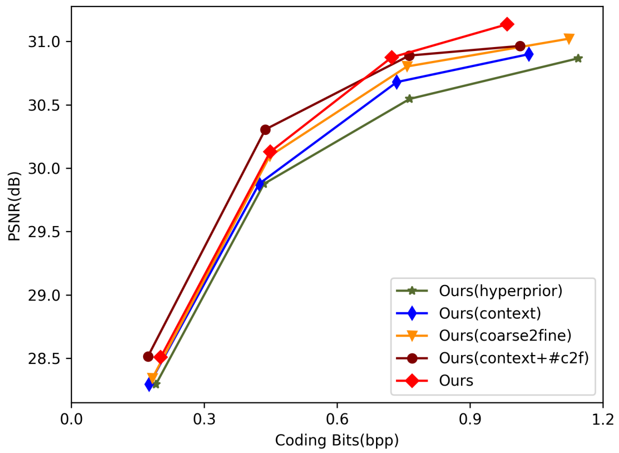


Fig. 11. The rate-distortion performance comparison of ablation results on different entropy estimation methods, including hyperprior [15], context [16], and coarse2fine [17] entropy models. All those entropy models share the same backbone.

original method, we can observe the quality (in terms of PSNR value) of the decoded images drops dramatically at the same bitrate, revealing the unsatisfactory capability of the redundancy removal led by the ablation of the transform module. Moreover, as shown in Fig. 9(b), the ablation of the transform module causes inferior performance on the PolyU dataset, where the noise type is unseen in the training phase, demonstrating the transform module plays a vital role in enhacing the generalization capability.

To better understand the performance of the transform module, in Fig. 10, we further visualize the FICs F_j^c (j∈$\{1, 4\}$) that are extracted from a noisy image by the transform modules T_j. More specifically, we rescale the FICs to the same spatial size for better visual comparison and the first 8 feature maps of F_1^c and F_4^c are exhibited in the Fig. 10 (a) and Fig. 10 (b), respectively. We can observe that the F_1^c contains rich structure and texture information while F_4^c is more complex and abstract, revealing more compact representations acquired in F_4^c. Moreover, the weak correlations among different channels of F_4^c demonstrate that the channel redundancy can be effectively eliminated by our transform module. In addition, compared with the FIC F_j^p (j∈$\{1, 4\}$) extracted from the pristine image (as shown in Fig. 10 (c) and Fig. 10 (d)), the similarities between F_j^p and F_j^c are significantly high, implying our proposed multi-scale transformation can disentangle the FIC component from the noisy image efficiently.

2) Entropy Model: In this subsection, we compare our entropy model with various widely-used entropy models, including the hyperprior entropy model [15], the context entropy model [16] and the coarse-to-fine entropy model [17]. For fair comparisons, we incorporate those entropy models in the same backbone and the results are shown in Fig. 11. We can observe the proposed entropy model estimates probability distribution of the latent representations more accurately. It should be noted that combining the context and our coarse-to-fine model (denoted as Ours(context+#c2f) in Fig. 11) achieves better results at the low bitrate, while the performance drops dramatically when the bitrate increases, since the context noise interferes the entropy estimation. In addition, the complexity of the context combined strategy is significantly high, while our entropy model is more efficient and outperforms the other entropy models consistently.

TABLE I

THE COMPARISON OF ENCODING AND DECODING TIMES (S).

Model	Encoding time (s)	Decoding time (s)
MRPNet-Coarse2Fine	0.241	0.395
MRPNet-VTM	98.688	0.106
MRPNet-Cheng	4.827	8.713
HyperPrior	0.041	0.034
Minnen *et al.*	4.750	8.713
Cheng *et al.*	4.804	8.922
Coarse2Fine	0.144	0.391
Ours	0.104	0.051

3) Loss function: The main components of our objective function include the triplet loss $\mathcal{L}_{trp}$, the noisy image reconstruction loss $\mathcal{L}_{nir}$ for feature disentanglement, and the perceptual loss $\mathcal{L}_{CR}$ for high-quality image decoding. In this sense, to verify the effectiveness of each component, the ablation studies are fairly conducted in this subsection. In particular, we first ablate the triplet loss $\mathcal{L}_{trp}$ at different coding bits. The results are shown in Fig. 9, where we can observe the rate-distortion performance of **Ours without $\mathcal{L}_{\mathbf{trp}}$** is inferior to the proposed original model both on synthetic noise and the real noise, demonstrating the $\mathcal{L}_{trp}$ has a positive effect both on the reconstruction accuracy and the generalization capability. Furthermore, we ablate the noisy image reconstruction loss $\mathcal{L}_{nir}$ and denote the results as **Ours (w/o $\mathcal{L}_{\mathbf{nir}}$)** in Fig. 9. Consistently, a significant performance drop can be observed. The reason lies in that the $\mathcal{L}_{nir}$ prevents the loss of content information during the feature disentanglement. Subsequently, we explore the importance of the perceptual loss $\mathcal{L}_{pc}$ by ablating it in the training phase, *i.e.* only the $\mathcal{L}_{mse}$ is used for the quality evaluation of the decoded image. The results are denoted as **Ours (w/o $\mathcal{L}_{\mathbf{pc}}$)** in Fig. 9. We can easily find that the PSNR values drop significantly at the high bitrate due to the fact that $\mathcal{L}_{pc}$ can suppress noise in the high-level feature space. Finally, we further ablate all components of objective function except for the $\mathcal{L}_{mse}$, R_1 and R_2 to study the collective effect of $\mathcal{L}_{trp}$, $\mathcal{L}_{nir}$ and $\mathcal{L}_{pc}$. The results are denoted as **Ours (only RD)** in Fig. 9, from which, we can observe the ablation model only performs well on the images corrupted by the synthesized noise at the low bitrate and hardly generalized well on the images distorted by real noise. In summary, from the above analyses, we can conclude that each component of our loss function serves a specific purpose in the proposed method.

D. Encoding and Decoding Complexity Analysis

Herein, we evaluate the encoding and decoding complexity on various compression schemes. The results are provided in Table I, which compares the running time of the encoder and decoder. It can be observed that the complexity of the proposed method is moderate which enjoys low encoding and decoding complexity compared with the most of the existing schemes. Moreover, the preprocessing then compression schemes may consume more time in encoding phase. It is also interesting to observe that the VTM has the lowest decoding complexity. However the associated encoding complexity is extremely high owing to the delicate rate distortion optimization in the encoding procedure. In addition, high encoding and decoding complexity can be seen in the context based entropy coding methods, such as the methods in Minnen *et al.* [16] and Cheng *et al.* [14]. The proposed method gently rises the encoding and decoding complexity when compared with the HyperPrior [15] scheme. Meanwhile, it is worthy to mention that the decoding speed of the proposed method is faster than the encoding speed owing to the adoption of an asymmetric encode-decode architecture, which significantly reduces network parameters and accelerates the inference procedure, revealing potential benefits in the real applications.

V. Conclusion

We have presented a novel image coding scheme for noisy images in the wild. The novelty of the proposed scheme lies in that the input image is distinctly represented with the latent representations which could be subsequently divided into the FIC and FAD, characterizing the intrinsic visual content and additive degradation information that does not need to be conveyed. The principled disentanglement scheme excels at removing the redundancy of spatial, channel and content, ensuring the handling of a wide variety of noisy images in the wild. The superiority of the proposed scheme is also demonstrated by the images with real-world noise and synthetic noise, as well as the pristine images, demonstrating the promising rate-distortion performance as well as the high generalization capability.

As one of the first attempts on this emerging topic, there are several limitations of the proposed method that could be improved in our future work. First, since the proposed method is based upon the statistics of natural images, it may not properly generalize to the graphical images and artwork. Developing new methodologies to identify noise in these types of images is worth further investigation. Second, currently the quality assessment of the compressed images is based upon the PSNR between the pristine and distorted images. In the future, how to faithfully evaluate the quality of the compressed in the wild images may be further exploited. Third, since the current method aims to preserve the intrinsic image content in the compression domain, the optimal combination with the pre-processing and post-processing strategies may also further improve the coding performance. This opens up new space for further exploration of efficiently compressing in the wild images.

References

[1] J. Ballé, V. Laparra, and E. P. Simoncelli, "Density modeling of images using a generalized normalization transformation," in *4th International Conference on Learning Representations, ICLR, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*. OpenReview.net.

[2] S. Ma, X. Zhang, C. Jia, Z. Zhao, S. Wang, and S. Wang, "Image and video compression with neural networks: A review," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 6, pp. 1683–1698, 2019.

[3] B. Li, S. Wang, and S. Wang, "Defending against noise by characterizing the rate-distortion functions in end-to-end noisy image compression," in *2021 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2021, pp. 3727–3731.

[4] Y. Chen, O. C. Au, and X. Fan, "Simultaneous map-based video denoising and rate-distortion optimized video encoding," *IEEE transactions on circuits and systems for video technology*, vol. 19, no. 1, pp. 15–26, 2008.

[5] J. Deng, A. Giladi, and F. G. Pancorbo, "Noise reduction prefiltering for video compression," *Standford University, Standford, CA, USA, Tech. Rep. EE398*, 2008.

[6] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in *Computer Vision - ECCV 2014, Zurich, Switzerland*, vol. 8692. Springer, 2014, pp. 184–199.

[7] W. Yang, R. T. Tan, S. Wang, Y. Fang, and J. Liu, "Single image deraining: From model-based to data-driven and beyond," *IEEE Transactions on pattern analysis and machine intelligence*, 2020.

[8] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising," *IEEE transactions on image processing*, vol. 26, no. 7, pp. 3142–3155, 2017.

[9] G. K. Wallace, "The jpeg still picture compression standard," *IEEE transactions on consumer electronics*, vol. 38, no. 1, pp. xviii–xxxiv, 1992.

[10] M. Rabbani and R. Joshi, "An overview of the jpeg 2000 still image compression standard," *Signal processing: Image communication*, vol. 17, no. 1, pp. 3–48, 2002.

[11] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, "Overview of the high efficiency video coding (hevc) standard," *IEEE Transactions on circuits and systems for video technology*, vol. 22, no. 12, pp. 1649–1668, 2012.

[12] B. Bross, Y.-K. Wang, Y. Ye, S. Liu, J. Chen, G. J. Sullivan, and J.-R. Ohm, "Overview of the versatile video coding (vvc) standard and its applications," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 10, pp. 3736–3764, 2021.

[13] F. Mentzer, G. D. Toderici, M. Tschannen, and E. Agustsson, "High-fidelity generative image compression," *Advances in Neural Information Processing Systems*, vol. 33, 2020.

[14] Z. Cheng, H. Sun, M. Takeuchi, and J. Katto, "Learned image compression with discretized gaussian mixture likelihoods and attention modules," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 7939–7948.

[15] J. Ballé, D. Minnen, S. Singh, S. J. Hwang, and N. Johnston, "Variational image compression with a scale hyperprior," in *6th International Conference on Learning Representations*, 2018.

[16] D. Minnen, J. Ballé, and G. D. Toderici, "Joint autoregressive and hierarchical priors for learned image compression," in *Advances in Neural Information Processing Systems*, vol. 31, 2018.

[17] Y. Hu, W. Yang, and J. Liu, "Coarse-to-fine hyper-prior modeling for learned image compression," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 07, 2020, pp. 11 013–11 020.

[18] B. K. Natarajan, "Filtering random noise from deterministic signals via data compression," *IEEE transactions on signal processing*, vol. 43, no. 11, pp. 2595–2605, 1995.

[19] W. Zhang, K. Ma, G. Zhai, and X. Yang, "Uncertainty-aware blind image quality assessment in the laboratory and wild," *IEEE Transactions on Image Processing*, vol. 30, pp. 3474–3486, 2021.

[20] ISO/IEC and ITU-T, "Final call for proposals for jpeg ai," *ISO/IEC JTC 1/SC29/WG1 N100095*, 2022.

[21] S. Motiian, M. Piccirilli, D. A. Adjeroh, and G. Doretto, "Unified deep supervised domain adaptation and generalization," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 5715–5725.

[22] Z. Xu, D. Liu, J. Yang, C. Raffel, and M. Niethammer, "Robust and generalizable visual representation learning via random convolutions," in *9th International Conference on Learning Representations, ICLR Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

[23] D. Li, Y. Yang, Y.-Z. Song, and T. M. Hospedales, "Deeper, broader and artier domain generalization," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 5542–5550.

[24] D. Kotovenko, A. Sanakoyeu, S. Lang, and B. Ommer, "Content and style disentanglement for artistic style transfer," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 4422–4431.

[25] A. Khosla, T. Zhou, T. Malisiewicz, A. A. Efros, and A. Torralba, "Undoing the damage of dataset bias," in *European Conference on Computer Vision, Florence, Italy*. Springer, 2012, pp. 158–171.

[26] P. Chattopadhyay, Y. Balaji, and J. Hoffman, "Learning to balance specificity and invariance for in and out of domain generalization," in *European Conference on Computer Vision, Glasgow, UK*. Springer, 2020, pp. 301–318.

[27] W. Du, H. Chen, and H. Yang, "Learning invariant representation for unsupervised image restoration," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 14 483–14 492.

[28] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *International conference on machine learning*. PMLR, 2020, pp. 1597–1607.

[29] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1. IEEE, 2005, pp. 539–546.

[30] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 815–823.

[31] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *European conference on computer vision, Amsterdam, The Netherlands*. Springer, 2016, pp. 630–645.

[32] F. Wang, M. Jiang, C. Qian, S. Yang, C. Li, H. Zhang, X. Wang, and X. Tang, "Residual attention network for image classification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 3156–3164.

[33] S. W. Zamir, A. Arora, S. Khan, M. Hayat, F. S. Khan, M.-H. Yang, and L. Shao, "Learning enriched features for real image restoration and enhancement," in *European Conference on Computer Vision, Glasgow, UK*. Springer, 2020, pp. 492–511.

[34] A. L. Maas, A. Y. Hannun, A. Y. Ng *et al.*, "Rectifier nonlinearities improve neural network acoustic models," in *Proc. icml*, vol. 30, no. 1. Citeseer, 2013, p. 3.

[35] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *European conference on computer vision, Amsterdam, The Netherlands*. Springer, 2016, pp. 694–711.

[36] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *3rd International Conference on Learning Representations, ICLR*, 2015.

[37] J. Liu, G. Lu, Z. Hu, and D. Xu, "A unified end-to-end framework for efficient deep image compression," *arXiv preprint arXiv:2002.03370*, 2020.

[38] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proceedings Eighth IEEE International Conference on Computer Vision*, vol. 2. IEEE, 2001, pp. 416–423.

[39] A. Abdelhamed, S. Lin, and M. S. Brown, "A high-quality denoising dataset for smartphone cameras," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[40] S. Nam, Y. Hwang, Y. Matsushita, and S. J. Kim, "A holistic approach to cross-channel image noise modeling and its application to image denoising," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1683–1691.

[41] J. Anaya and A. Barbu, "Renoir - a dataset for real low-light image noise reduction," *Journal of Visual Communication and Image Representation.*, vol. 51, pp. 144–154, 2018.

[42] J. Xu, H. Li, Z. Liang, D. Zhang, and L. Zhang, "Real-world noisy image denoising: A new benchmark," *arXiv preprint arXiv:1804.02603*, 2018.

[43] E. Kodak, "Kodak lossless true color image suite (photocd pcd0992)," *URL http://r0k. us/graphics/kodak*, vol. 6, 1993.

[44] T. Plotz and S. Roth, "Benchmarking denoising algorithms with real photographs," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1586–1595.

[45] A. Foi, M. Trimeche, V. Katkovnik, and K. Egiazarian, "Practical poissonian-gaussian noise modeling and fitting for single-image raw-data," *IEEE Transactions on Image Processing*, vol. 17, no. 10, pp. 1737–1754, 2008.

[46] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization." *ICLR*, vol. 9, 2015.

[47] "VVC software vtm-14.0," https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware_VTM/-/tags/VTM-14.0, online; accessed 12 December 2021.

[48] S. W. Zamir, A. Arora, S. Khan, M. Hayat, F. S. Khan, M.-H. Yang, and L. Shao, "Multi-stage progressive image restoration," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.