An End-to-End Joint Learning Scheme of Image Compression and Quality Enhancement with Improved Entropy Minimization

Jooyoung Lee\textsuperscript{1,3}, Seunghyun Cho\textsuperscript{2}, and Munchurl Kim\textsuperscript{*3}

\textsuperscript{1} Electronics and Telecommunications Research Institute, Korea
leejy1003@etri.re.kr

\textsuperscript{2} Kyungnam University, Korea
scho@kyungnam.ac.kr

\textsuperscript{3} Korea Advanced Institute of Science and Technology, Korea
mkimee@kaist.ac.kr

Abstract. Recently, learned image compression methods have been actively studied. Among them, entropy-minimization based approaches have achieved superior results compared to conventional image codecs such as BPG and JPEG2000. However, the quality enhancement and rate-minimization are conflictively coupled in the process of image compression. That is, maintaining high image quality entails less compression and vice versa. However, by jointly training separate quality enhancement in conjunction with image compression, the coding efficiency can be improved. In this paper, we propose a novel joint learning scheme of image compression and quality enhancement, called JointIQ-Net, as well as entropy model improvement, thus achieving significantly improved coding efficiency against the previous methods. Our proposed JointIQ-Net combines an image compression sub-network and a quality enhancement sub-network in a cascade, both of which are end-to-end trained in a combined manner within the JointIQ-Net. Also the JointIQ-Net benefits from improved entropy-minimization that newly adopts a Gaussian Mixture Model (GMM) and further exploits global context to estimate the probabilities of latent representations. In order to show the effectiveness of our proposed JointIQ-Net, extensive experiments have been performed, and showed that the JointIQ-Net achieves a remarkable performance improvement in coding efficiency in terms of both PSNR and MS-SSIM, compared to the previous learned image compression methods and the conventional codecs such as VVC Intra (VTM 7.1), BPG, and JPEG2000. To the best of our knowledge, this is the first end-to-end optimized image compression method that outperforms VTM 7.1 (Intra), the latest reference software of the VVC standard, in terms of the PSNR and MS-SSIM.

Keywords: end-to-end image compression, entropy minimization, image quality enhancement
Fig. 1: Rate-distortion curves of the proposed method and competitive methods for the Kodak PhotoCD image dataset [16]. The left and right plots represent RD-curves in terms of (a) PSNR and (b) MS-SSIM, respectively. Note that the measured MS-SSIM values are presented in the unit of decibels as in the previous works [7,20,18] to better distinguish the performance differences.

1 Introduction

Recently, significant progress in artificial neural networks has led to many ground-breaking achievements in various research fields. In image and video compression domain, a number of learning based studies [25,12,6,24,7,18,20,21,19] have been conducted. Especially, some latest end-to-end optimized image compression approaches [18,20] based on entropy minimization have already shown better compression performance than those of the existing image compression codecs such as BPG [8] and JPEG2000 [23], despite a short history of the field. The basic approach to entropy minimization is to train analysis (encoder) / synthesis (decoder) transform networks allowing them to reduce entropy of transformed latent representations, keeping the quality of reconstructed images as close as possible to the originals. Entropy minimization approaches can be viewed from two different aspects: prior probability modeling and context exploitation. Prior probability modeling is a main element of entropy minimization and allows an entropy model to approximate the actual entropy of latent representations, which plays a key role for both training and actual entropy coding/decoding. For each transformed representation, an image compression method estimates the parameters of the prior probability model, based on contexts such as previously decoded neighbor representations or some bit-allocated side informations. A better context can be regarded as the information given to the model parameter estimator, which help predict the distributions of latent representations more precisely.

The latest two entropy minimization approaches [18,20] achieved noticeable compression performance, but their methods focused on building new entropy models with context exploitation in an autoregressive manner, rather than utilizing the up-to-date architectural techniques. Meanwhile, in the field of quality enhancement, a number of studies have been continuously conducted in an architectural perspective and have shown superior results compared to the traditional methods, as described in Section 2. However, there has been few work on jointly taking into account both image compression and quality enhancement in a unified architecture, although worthwhile to restore...
Fig. 2: Comparison of sample reconstruction images including the ground truth, our method, Lee et al. [18]'s approach, Ballé et al. (2018) [7]'s approach, BPG [8], and VVC Intra (VTM 7.1) [1].

the compressed images with coding artifacts as close as possible to the quality of uncompressed input. Therefore, in this paper, we propose a novel joint learning scheme that incorporates quality enhancement and image compression so as to allow them to collaborate each other for higher coding efficiency. However, we do not propose any specific quality enhancement network to be incorporated into our new image compression network. Instead, an state-of-the-art (SOTA) method is adopted, so that any advanced method can be combined with our image compression network in the proposed unified joint learning architecture.

In addition, we present a novel image compression network that incorporates an improved entropy minimization method with GMM-based prior probability modeling and global context exploitation. In terms of prior probability modeling, we adopt a more generalized form with a GMM. The GMM was simply mentioned but was not used in Minnen et al. [20]'s approach where a single Gaussian model was used in their formulation and experiments. In our prior probability modeling, the sub-network for estimating the model parameters of GMM is trained in the course of jointly learning image compression and quality enhancement, yielding the improved estimation accuracy for them.

From a contextual perspective, we define a new type of global context for entropy minimization in an autoregressive manner. The autoregressive approaches [18,20] estimate the distribution of a current latent representation using its adjacent known representations, thus leading to improve the compression performance by removing the correlations between the current latent representation and its neighbors. Although their methods effectively remove the spatial and inter-channel correlations among the transformed representations, our global context exploitation further improves the coding efficiency by removing the remaining spatial correlations across a wider area of each input image, which has been motivated by the known wisdom [10,17] that exploits self-similarity in input images.

Fig. [1] shows the coding efficiency curves for our JointIQ-Net, Versatile Video Coding (VVC) Intra (VTM 7.1 [1]), BPG [8], JPEG2000 [23] and the deep learning based
SOTA methods [18,7,24]. As shown in Fig. 1-(a) and -(b), our JointIQ-Net outperforms all the image compression methods in terms of both PSNR and MS-SSIM. It is noted that our JointIQ-Net is the first deep image compression scheme that surpasses the VVC Intra coding. Fig. 2 visually compares some reconstructed images for our JointIQ-Net and the existing methods [18,7,8,23]. For similar compression ratios, it is clearly shown in Fig. 2 that our JointIQ-Net yields the reconstructed images of higher quality in both quantitative measures (PSNR) and qualitative viewing. The key contributions of our work are as follows:

- We first propose a novel end-to-end learning scheme, called JointIQ-Net, that can jointly optimize both image compression and quality enhancement;
- To the best of our knowledge, the JointIQ-Net is the first deep image compression scheme that outperforms in terms of both PSNR and MS-SSIM the VVC Intra coding (VTM 7.1 [1]) which has been almost finalized for standardization by ISO/IEC MPEG and ITU-T VCEG, also yielding significant improvements over BPG, JPEG2000, and the learned SOTA image compression approaches.
- We propose an improved entropy-minimization method that uses a GMM for prior probability modeling, whose parameters are accurately estimated by the improved estimator trained in the joint optimization of image compression and quality enhancement, yielding the improved coding efficiency;
- To further improve the entropy-minimization method, we utilize global context in estimation of the GMM parameters, which captures a wider context information and helps reduce the spatial correlations between a current latent representation and its neighbors in a non-local extent;

2 Related work

Artificial-neural-network (ANN) based image compression approaches can be divided into two folds: First, some approaches [25,12] try to achieve a small number (or ratio) of latent representations while maintaining the original information as much as possible in latent spaces. Based on this concept, Toderici et al. [25] introduced a novel image compression method using a fixed number of latent binary representations, which improve the image quality in an progressive manner. Then Johnston et al. [12] enhanced the network operation method of Toderici et al.’s network to achieve better coding efficiency; Second, some other approaches [6,24,17,18,20] minimize the entropy of the latent representations, which transforms them to have low entropy to be represented in a small number of bits by using their own entropy models. Ballé et al.(2017) [6] and Theis et al. [24] introduced a new image compression method based on entropy minimization. Ballé et al.(2018) [7] enhanced the entropy model by adopting a hierarchical prior model for estimating standard deviations of the latent representations in an input-adaptive manner, whereas the first two approaches [6,24] train their image compression networks with their prior model parameters fixed during inference. Minnen et al. [20] and Lee et al. [18] utilize adjacent regions of known latent representations as additional contexts for the parameter estimation of prior models, based on the idea that entropy-coding and decoding process can be conducted in an autoregressive manner (e.g. a
raster scanning order) and spatially adjacent representations tend to have high correlations. Both approaches enhanced the compression performance and obtained better results than BPG [8] that is the image compression codec based on HEVC (ISO/IEC 23008-2, ITU-T H.265) [11].

Meanwhile, ANN-based image restoration, such as super resolution (SR) and denoising, has become an indispensable method by far surpassing handcrafted algorithms. Kim et al. [14]’s approach, first introduced a deep network architecture based on residual learning for SR, named VDSR, and obtained substantial boost in SR performance. Zhang et al. [27]’s approach has achieved further improvement by exploiting residual dense blocks (RDBs), each of which comprises densely connected convolutional layers and a local skip connection. Kim et al. [13]’s approach, grouped residual dense network (GRDN), has extended the previous work by grouping multiple RDBs, named grouped residual dense blocks (GRDBs), and arranged the multiple GRDBs in the network. Furthermore, they incorporate a more deeper architecture that allows the convolutional layers to process down-scaled representations, and also adopt spatial and channel-wise attention layers. Based on this enhanced architecture, they have obtained the state-of-the-art performance in the image denoising task. Recently, Cho et al. [9]’s approach has utilized GRDN [13] for reducing artifacts caused by a new image codec, which is the intra coding of VVC [5], under standardization, and they have achieved a noticeable quality improvement. However, GRDN in Cho et al. [9]’s approach has been separately optimized against the image codec.

3 Proposed network architecture

3.1 Joint learning scheme of image compression and quality enhancement

Fig. 3 shows our end-to-end joint learning scheme, JointIQ-Net, of image compression and quality enhancement in cascade. As mentioned, in this paper, we propose a novel image compression network but adopt an existing image quality enhancement network for the JointIQ-Net. Consequently, the proposed architecture provides high flexibility and extensibility. In particular, our method can easily accommodate future’s advanced image quality enhancement networks, and it also allows various combinations of image compression and quality enhancement methods. That is, separately developed image compression networks and quality enhancement networks can easily be combined and can jointly be optimized in a unified architecture by minimizing the following total loss:

$$\mathcal{L} = R + \lambda D(x, Q(I(x)))$$  (1)
where $I$ is an image compression with input $x$, and $Q$ is a quality enhancement function with input $\hat{x} = I(x)$ which is an intermediate reconstruction output of $I$, $R$, $D$, and $\lambda$ represent the rate, distortion, and a balancing parameter, respectively. In contrast to the previous methods [6,24,7,20,18] that train the image compression networks, $I$, to reconstruct the output images with as small distortion as possible, we regard the outputs of $I$ in Eq. (1) as an intermediate latent representation, $\hat{x}$, which is fed into the quality enhancement sub-network $Q$. So, the distortion $D$ is measured between the input $x$ and the final output $x' = Q(\hat{x})$ reconstructed by $Q$. Consequently, our architecture allows two sub-networks to be jointly optimized towards minimizing the total loss Eq. (1). Note that $\hat{x}$ is best represented in a sense that $Q$ outputs the final reconstruction with high fidelity.

It should be noted that our work is not intended to propose an customized quality enhancement network, but to present an joint end-to-end learning scheme of both image compression and quality enhancement. Thus, to choose an appropriate quality enhancement network for our experiments, we combine a reference image compression method [18] with various quality improvement methods, VDSR [14], RDN [27] and GRDN [13], in cascade connections. For fair comparisons, the numbers of parameters and layers for the quality enhancement networks are adjusted to have similar computation complexities. Fig. 4 show the coding efficiency results for the combined image compression and quality enhancement networks. The experimental results are obtained by measuring average PSNR or MS-SSIM values over the Kodak PhotoCD image dataset [16]. In the supplementary material, we also provide the test results over the CLIC [2] validation imageset and Tecnick [4] imagesets. As shown in Fig. 4, the GRDN [13] yields the highest compression performance in combination with the image compression method. So, we use GRDN [13] for our JointIQ-Net.

To verify the effectiveness of our joint learning scheme of image compression and quality enhancement, we compare two cascaded versions of the reference image compression model [18] and GRDN [13]. The first cascaded version is optimized for image compression and quality enhancement in an end-to-end manner, whereas the reference image compression model and the GRDN are separately learned in the second cascaded
version where the GRDN is trained with the outputs of a separately trained reference image compression network for the same training dataset. Fig. 4-(b) shows the PSNR performance for the two cascaded versions. As shown, the first cascaded version outperforms the seconded version, especially in higher bit-rate ranges. The jointly trained GRDN in the first cascaded version effectively works over the whole bit-rate range while the separately trained GRDN can restore the visual quality of reconstructed images in low bit-rate ranges but improves less in high bit-rate ranges.

It might be viewed that the quality enhancement network can be thought of as a decoder part of the image compression network. It can also be thought that increasing the decoder complexity of the image compression network might bring a similar amount of coding efficiency improvement instead of cascading the quality enhancement network. However, the purpose of our work is targeted for a flexible joint learning scheme of image compression and any image quality enhancement solution that can be independently developed outside the image compression. Simply increasing the decoder part complexity may not bring a meaningful coding efficiency improvement due to its limited structure.

3.2 Proposed image compression network

The overall network architecture of JointIQ-Net is illustrated in Fig. 5. As mentioned in section 3.1, our image compression network is connected with GRDN, adopted as a quality enhancement sub-network, in a cascade. The image compression network of the proposed JointIQ-Net is based on the existing approach [18]. Therefore, we basically use the same rate-distortion optimization framework and transform functions. The JointIQ-Net transforms input $x$ into latent representations $y$, and $y$ is then quantized into $\hat{y}$. In addition, we also use the hyperprior $\hat{z}$, proposed in Ballé et al. (2018) [7]s approach, which further captures spatial correlations of $\hat{y}$. Accordingly, we use four...
fundamental transform functions: an analysis transform \( g_a(x; \phi_g) \), a synthesis transform \( g_s(\hat{y}; \theta_g) \), an analysis transform \( h_a(\hat{z}; \phi_h) \), and a synthesis transform \( h_s(\hat{z}; \theta_h) \), as in the previous methods [18]. The optimization process ensures the JointIQ-Net to yield the entropy of \( \hat{y} \) and \( \hat{z} \) as low as possible and also to yield \( x' \), reconstructed from \( \hat{y} \), as close to the original visual quality as possible. To allow this rate-distortion optimization, along with the distortion between the input \( x \) and output \( x' \), the rate is calculated based on the prior probability models for \( \hat{y} \) and \( \hat{z} \). For \( \hat{z} \), we use a simple zero-mean Gaussian model convolved with \( \mathcal{U}(-\frac{1}{2}, \frac{1}{2}) \), whose standard deviation values are found from training, whereas the parameters of the prior probability model for \( \hat{y} \) are estimated by the model parameter estimator \( f \) in an autoregressive manner as in the previous method [18].

The model parameter estimator \( f \) in the previous method [18] utilizes the two types of contexts, \( c' \) reconstructed from the hyperprior \( \hat{z} \) and \( c'' \) extracted from the adjacent known representations of \( \hat{y} \). In addition, we let \( f \) additionally utilize a global context, denoted as \( c''' \), for estimating the model parameters more precisely as described in Section 4.3. The functions to extract the three types of contexts are denoted as \( E' \), \( E'' \), and \( E''' \), respectively. With the three given contexts, \( f \) estimates the parameters of GMM (convolved with \( \mathcal{U}(-\frac{1}{2}, \frac{1}{2}) \)), adopted as a prior probability model for \( \hat{y} \) in our method, as described in Section 4.4. This parameter estimation is used in the entropy coding and decoding processes, represented as EC and ED, as well as in the calculation of the rate term for training. In addition, we enhance the structure of the model estimator \( f \), based on Lee et al. [18]’s method, by extending it to a new model estimator. The new model estimator incorporates a model parameter refinement module (MPRM) to improve the capability of model parameter estimation, as shown in Fig. 5. The MPRM has two residual blocks, each of which contains the fully-connected layers and the corresponding non-linear activation layers.

4 Improved entropy models and parameter estimation for entropy-minimization

The previous entropy-minimization methods [18,20] utilize local contexts to estimate the prior model parameters for each \( \hat{y}_i \). For this, they utilize the neighbor latent representations of a current representation, \( \hat{y}_i \), for estimating \( \mu_i \) and \( \sigma_i \) of a single Gaussian prior model (convolved with a uniform function) for \( \hat{y}_i \). These approaches have two limitations: (i) A single Gaussian model has a limited capability to model various distributions of latent representations. In this paper, we use a Gaussian mixture model GMM; (ii) Extracting the context information from neighbor latent representations is limited when their correlations widely exist over the entire spatial domains.

4.1 Gaussian mixture model (GMM) for prior distributions

The existing autoregressive methods [18,20] use the single Gaussian distribution to model the distribution of each \( \hat{y}_i \). Although their transform networks can produce the latent representations that follow single Gaussian distributions, such a single Gaussian modeling is limited in predicting the actual distributions of latent representations, thus
leading to sub-optimal performance. Instead, we use a more generalized form, GMM, of a prior probability model.

4.2 Formulation for Entropy Models

We basically use the same R-D optimization framework as the existing approaches [18,20]. The objective function includes the rate and distortion terms, as shown in Eq. 2, and the parameter $\lambda$ is used to adjust the balance between the rate and distortion in the optimization process:

$$
\mathcal{L} = R + \lambda D
$$

The rate term is composed of the cross-entropy for $\hat{z}$ and $\hat{y}|\hat{z}$. To deal with the discontinuities due to quantization, as in the previous methods [7,18,20], a density function convolved with a uniform function $U\left(-\frac{1}{2}, \frac{1}{2}\right)$ is used for approximating probability mass function (PMF) of $\hat{y}$. Correspondingly, for training, the noisy representations $\tilde{y}$ and $\tilde{z}$ following uniform distributions whose mean values are $y$ and $z$, respectively, are used to fit the actual sample distributions to the PMF-approximating functions. To model the distributions of $\tilde{z}$, as in previous approach [18], we simply use zero-mean Gaussian density functions (convolved with a uniform density function), whose standard deviations are optimized via training. Whereas, we extend the entropy model for $\tilde{y}|\hat{z}$ based on a GMM as:

$$
p_{\tilde{y}|\hat{z}}(\tilde{y}|\hat{z}) = \prod_{i} \left( \sum_{g=1}^{G} \pi_{i,g} N(\mu_{i,g}, \sigma_{i,g}^{2}) * U\left(-\frac{1}{2}, \frac{1}{2}\right) \right)(\tilde{y})
$$

where $G$ is the number of Gaussian distribution functions. The distribution estimator $f$ predicts $3 \times G$ parameters so that each of the $G$ Gaussian distributions has its own weight, mean, and standard deviation parameters, denoted as $\pi_{i,g}$, $\mu_{i,g}$, and $\sigma_{i,g}$, respectively. The mean squared error (MSE) is basically used as the distortion term for optimization in Eq. 2 and we additionally provide experimental results of the MS-SSIM [26] optimized models.

4.3 Global context for model parameter estimation

In order to better extract context information for the current latent representation $y_i$, we can use a global context by aggregating all possible contexts from the whole area of known representations to estimate the prior model parameters. For this, we define the global context as information aggregated from both local and non-local context regions, where the local context region is within the fixed distance, denoted as $K$, from
the current representation \( y_{h} \), and the non-local region is the whole causal area outside the local context region. Fig. 6 shows an example of the local and non-local context regions.

As global context \( c_{i}'''' \), we use a weighted mean value and a weighted standard deviation value aggregated from the global context region, which is the whole known spatial area within a channel of \( \hat{y} \). We obtain the global context \( c_{i}'''' \) from \( \hat{y} \), which is a linearly transformed version of \( \hat{y} \) via an \( 1 \times 1 \) convolution layer, to capture the correlations across the different channels of \( \hat{y} \) as well. Specifically, The global context \( c_{i}'''' = \{ \mu_{i}^{*}, \sigma_{i}^{*} \} \) consists of a weighted mean \( \mu_{i}^{*} \) and a weighted standard deviation \( \sigma_{i}^{*} \), both of which are defined as:

\[
\mu_{i}^{*} = \sum_{k,l \in S} w_{k,l}^{(i)} \hat{y}_{k,i} \quad (4)
\]

\[
\sigma_{i}^{*} = \sqrt{\frac{\sum_{k,l \in S} w_{k,l}^{(i)} (\hat{y}_{k,i}^{(i)} - \mu_{i}^{*})^{2}}{1 - \sum_{k,l \in S} w_{k,l}^{(i)^{2}}}} \quad (5)
\]

where \( i = [i_{c}, i_{h}, i_{v}] \) is a 3-d spatio-channel-wise position index indicating a current position \((i_{h}, i_{v})\) in the \( i_{c}\)-th channel. \( w_{k,l}^{(i)} \) is a weight variable for the relative coordinates \((k, l)\) based on the current location \((i_{h}, i_{v})\), and \( \hat{y}_{k,i}^{(i)} \) is a representation of \( \hat{y}^{(i)} \).
at location \((i_h-k, i_v-l)\), within the global context region \(S\). \(\hat{y}^{(i)}\) is the two-dimensional representations within \(i_{-th}\) channel of \(\hat{y}\). The weight variables in \(w^{(i)}\) are the normalized weights that are element-wise multiplied to \(\hat{y}^{(i)}\) for the weighted mean in Eq. 4 and to the difference squares of \((\hat{y}_{i_h-k, i_v-l} - \mu_i^*)\) in Eq. 5.

Here, the key issue is to find an optimal set of the weight variables \(w^{(i)}\) at every location \(i\). To obtain \(w^{(i)}\) from a fixed number of trainable variables \(\psi^{(i)}\), \(w^{(i)}\) is estimated based on a 2-dimensional extension to the 1-dimensional global context extraction scheme of Shaw et al. [22]’s approach. Fig. 6 shows the global context region that consists of the local context region within a fixed distance \(K\), which is covered by the trainable variables \(\psi^{(i)}\), and the non-local context region of a variable size, outside of the local context region. In the global context extraction, the non-local context region becomes enlarged as the local context window that defines the local context area slides over a feature map, thus increasing the number of weights \(w^{(i)}\). To deal with the non-local context region which cannot covered by a fixed size of trainable variables \(\psi^{(i)}\), a variable of \(\psi^{(i)}\) allocated to the nearest local context area is used for each spatial position within the non-local context region, as shown in Fig. 6. As a result, we can obtain a set of trainable variables \(\psi^{(i)}\), denoted as \(a^{(i)}\), corresponding to the global context region. Then \(w^{(i)}\) is calculated by normalizing \(a^{(i)}\), via softmax as follows:

\[
w^{(i)} = \text{softmax}(a^{(i)})
\]

where \(a^{(i)} = \{\psi^{(i)}_{\text{clip}(k,K),\text{clip}(l,K)}|k,l \in S\}\) and \(\text{clip}(x,K) = \max(-K, \min(K,x))\).

Note that \(\psi^{(i)}_{k,l} = \psi^{(i)}_{k+1,l}^{(i+1)}\) within the same channel (over the same spatial feature space). Fig. 7 visualizes the trained \(\psi^{(i)}\) examples for several channels of \(\hat{y}\). Fig. 7-(a) shows the case that the context of the channel is dependent of the neighbor representations just next to the current latent representation while Fig. 7-(d) shows the case that the context of the channel is dependent of the widely spread neighbor representations.

## 5 Implementation

The detailed structure of our JointIQ-Net is depicted in Fig. 5 where \(N\) and \(M\) are set according to \(\lambda\) values. The values of \(N\) and \(M\) for different \(\lambda\) values are tabulated in Table 1. We set \(G\), the number of Gaussian pdfs for each prior distribution, to 3. Therefore, the model parameter estimator \(f\) outputs \(9 \times M\) values for \(M\) representations of \(\hat{y}_i\), which are located at a specific spatial position of \(\hat{y}\). For obtaining the global contexts, we set \(K\) to 7, and we utilize the global contexts only when the number of representations in the global context region is 30 or higher, in order to maintain statistical significance of the global contexts. For less than 30 representations, we set the global contexts to all zeros. For the GRDN in our final model, we set the number of GRDBs, RDBs in each GRDB, the number of convolutional layers in each RDB, and the number of kernels used by each convolutional layer to 4, 4, 8, and 64, respectively. Note that for the GDRN used in Fig. 4 and 8 is a light-weight version for which the above parameters are set to 4, 3, 3, and 32, respectively, for simulation at low complexity.

In the training phase, we used 51,140 256×256 patches extracted from CLIC [2] train imageset. The mini-batch size is set to 8, and all the models are trained using the
ADAM optimizer [15] with its default setting. The models were trained using their own initial learning rates in Table 1. We applied the gradient decaying by reducing the learning rate to half at every 50,000 steps during the final 300,000 steps. For proper scaling of the $\lambda$ range, Eq. 7 is used as an objective cost function in the actual implementation.

For training the final models in Fig. 1 which include the deeper $Q$ (GRDN) sub-networks, we used the same $256 \times 256$-sized patches as a training set, but we randomly extracted $96 \times 96$-sized patches from the outputs of the $I$ sub-network, and fed them into the $Q$ sub-network. The distortion term was calculated against the corresponding area of input patches, and the rate term was also calculated over the corresponding $6 \times 6$-sized area out of the $16 \times 16$-sized spatial area of $\hat{y}$. To reduce the training time, we utilized the pre-trained $I$ sub-networks of the models with the lightweight GRDNs. We first train the $Q$ sub-network using only the distortion term for 100K iterations, and then further optimize the $I$ and $Q$ sub-networks in an end-to-end manner for additional 1M iterations.

$$ L = \frac{\lambda}{W_y \cdot H_y \cdot 256} R + \frac{1 - \lambda}{1000} D. $$

(7)

| $\lambda$ | $N$  | $M$  | No. of iterations | Initial learning rates |
|---------|------|------|--------------------|-----------------------|
| 0.5     | 128  | 128  | 1.2M               | 1e-4                  |
| 0.35    | 128  | 128  | 1.2M               | 1e-4                  |
| 0.23    | 128  | 192  | 1.5M               | 1e-4                  |
| 0.12    | 192  | 256  | 1.5M               | 1e-4                  |
| 0.06    | 192  | 420  | 2.0M               | 5e-5                  |
| 0.03    | 192  | 420  | 2.0M               | 5e-5                  |
| 0.017   | 256  | 600  | 3.0M               | 5e-5                  |
| 0.01    | 256  | 600  | 3.0M               | 3e-5                  |

6 Experiments

6.1 Experimental environments

To verify the performance of the proposed method, we measured the average bits per pixel (BPP) and quality of the reconstructed images over the Kodak PhotoCD image dataset [16]. The PSNR and MS-SSIM metrics are used to measure the quantitative qualities. For each quality metric, eight models were trained with different $\lambda$ values, and we evaluated them by comparing the resulting R-D curve with those of the existing ANN-based approaches, such as Lee et al. [18], Ballé et al. (2018) [7], and Theis et al. [24], and the conventional codecs, such as VVC Intra (VTM 7.1 [1]), BPG [8], and
6.2 Experimental results

We compared the compression performance of our method with those of the other existing approaches using the rate-distortion curves, in terms of PSNR and MS-SSIM. As demonstrated in Fig. 1, our method outperforms all the previous methods under comparison in terms of both PSNR and MS-SSIM. Specifically, the compression gains are obtained with 1.65 (48.40)%, 16.96 (14.83)%, 26.58 (26.65)%, 22.57 (57.35)%, and 45.48 (73.65)% in the BD-rates of PSNR (MS-SSIM) over VVC Intra (VTM 7.1 [1]), Lee et al. [18]'s method, Ballé et al. (2018) [7]'s method, BPG [8] and JPEG2000 [23], respectively. In the supplementary material, we provide the examples of reconstructed images with those of the other methods.

6.3 Ablation study

In order to verify the effectiveness of each proposed element, we conducted the ablation study as follows: We excluded each proposed element from the full model, and trained the models in the same way as in the experiments of Section 6. We compared the test results of each model with those of the full model. Four different models were evaluated and their excluded components are GRDN [13], global context (GC), model parameter refinement module (MPRM), and Gaussian mixture model (GMM), respectively. Note that, the global context is also excluded when excluding MPRM because the global context is processed by the MPRM in our full model. In addition, as a baseline, the compression performance of Lee et al. [18]'s method is also included in the comparison. Fig. 8 (a) shows the PSNR performances for various versions of our model used in the ablation study. As shown in Fig. 8 (a), when GRDN [13] is excluded, significant performance degradation occurs. This indicates that the proposed joint learning scheme can play an important role in improving compression performance. The global context also
improves performance, but the amount of PSNR performance improvement is relatively low. When both MPRM and global context are excluded, the performance degradation becomes more noticeable. The results also show that MPRM has a greater impact in a higher bpp range. Whereas, when we use a single Gaussian model instead of a GMM, a similar level of performance degradation occurs over the entire bpp range. Table 2 compares the quantitative results between the final model and each of the element-excluded models in terms of BD-rate loss.

To see the effectiveness of a GMM prior model, we performed a comparison test between the reference model using a single Gaussian prior model and the same model with a GMM prior model, without the other architectural or contextual changes. As shown in Fig. 8, we’ve obtained 3.63% of coding gain when using the GMM prior model compared to the reference model. Note that two networks have the same architecture, except for the number of the output nodes of the model estimator $f$.

| Models            | BD-rate losses (%) |
|-------------------|--------------------|
| without GRDN      | 8.61               |
| without GC        | 0.92               |
| without GC/MPRM   | 3.71               |
| without GMM       | 2.76               |

### 7 Conclusion

In this paper, we proposed a new image compression method, called JointIQ-Net, that outperforms VVC Intra (VTM 7.1), BPG, JPEG2000, and the state-of-the-art ANN-based image compression approaches. To the best of our knowledge, our JointIQ-Net is the first learned image compression approach that surpasses the VVC Intra coding, in terms of both PSNR and MS-SSIM. For improving the coding efficiency of the JointIQ-Net, we have built a new joint learning scheme that incorporates both image compression and quality enhancement, allowing them to be end-to-end optimized in a unified manner. From the perspective of image compression, we have improved the entropy model by adopting GMM as a more generalized prior probability model for the transformed representations. In addition, we have enhanced the capability of the model parameter estimation by utilizing the global contexts that can reduce the remaining correlations over the global regions of the transformed representations.
References

1. Versatile video coding reference software version 7.1 (VTM-7.1) (December 2019), https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware_VTM/tags/VTM-7.1

2. Workshop and challenge on learned image compression (2019), https://www.compression.cc/

3. Agustsson, E., Mentzer, F., Tschannen, M., Cavigelli, L., Timofte, R., Benini, L., Gool, L.V.: Soft-to-hard vector quantization for end-to-end learned compression of images and neural networks. In: Advances in Neural Information Processing Systems 30. pp. 1141–1151 (2017)

4. Asuni, N., Giachetti, A.: TESTIMAGES: a Large-scale Archive for Testing Visual Devices and Basic Image Processing Algorithms. In: Giachetti, A. (ed.) Smart Tools and Apps for Graphics - Eurographics Italian Chapter Conference. The Eurographics Association (2014). https://doi.org/10.2312/stag.20141242

5. B. Bross, J. Chen, S.L.: Versatile video coding (draft 5). Draft, JVET (2019)

6. Ballé, J., Laparra, V., Simoncelli, E.P.: End-to-end optimized image compression. In: the 5th Int. Conf. on Learning Representations (2017), http://arxiv.org/abs/1611.01704

7. Ballé, J., Minnen, D., Singh, S., Hwang, S.J., Johnston, N.: Variational image compression with a scale hyperprior. In: the 6th Int. Conf. on Learning Representations (2018), http://arxiv.org/abs/1802.01436

8. Bellard, F.: Bpg image format (2014), http://bellard.org/bpg/

9. Cho, S., Lee, J., Kim, J., Kim, Y.: Low bit-rate image compression based on post-processing with grouped residual dense network. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops (June 2019)

10. Glasner, D., Bagon, S., Irani, M.: Super-resolution from a single image. In: IEEE International Conference on Computer Vision (ICCV). pp. 349–356 (2009)

11. Information technology – high efficiency coding and media delivery in heterogeneous environments – part 2: High efficiency video coding. Standard, ISO/IEC (2013)

12. Johnston, N., Vincent, D., Minnen, D., Covell, M., Singh, S., Chinen, T., Jin Hwang, S., Shor, J., Toderici, G.: Improved lossy image compression with priming and spatially adaptive bit rates for recurrent networks. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2018)

13. Kim, D.W., Chung, J.R., Jung, S.W.: Grdn:grouped residual dense network for real image denoising and gan-based real-world noise modeling. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops (2019)

14. Kim, J., Lee, J.K., Lee, K.M.: Accurate image super-resolution using very deep convolutional networks. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR Oral) (June 2016)

15. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: the 3rd Int. Conf. on Learning Representations (2015), http://arxiv.org/abs/1412.6980

16. Kodak, E.: Kodak lossless true color image suite (photocd pcd0992) (1993), http://r0k.us/graphics/kodak/

17. Lee, D.Y., Lee, J., Choi, J.H., Jong-Ok, K., Kim, H.Y., Soo, C.J.: Gpu-based real-time super-resolution system for high-quality uhd video up-conversion. In: The Journal of Supercomputing. vol. 65(3) (September 2013). https://doi.org/10.1007/s11227-017-2136-1

18. Lee, J., Cho, S., Beack, S.K.: Context-adaptive entropy model for end-to-end optimized image compression. In: the 7th Int. Conf. on Learning Representations (May 2019)

19. Lu, G., Ouyang, W., Xu, D., Zhang, X., Cai, C., Gao, Z.: Dvc: An end-to-end deep video compression framework. arXiv preprint arXiv:1812.00101 (2018)
20. Minnen, D., Ballé, J., Toderici, G.: Joint autoregressive and hierarchical priors for learned image compression. In: Advances in Neural Information Processing Systems (May 2018)
21. Park, W., Kim, M.: Deep predictive video compression with bi-directional prediction. CoRR abs/1904.02909 (2019), http://arxiv.org/abs/1904.02909
22. Shaw, P., Uszkoreit, J., Vaswani, A.: Self-attention with relative position representations. In: Proc. of NAACL (2018)
23. Taubman, D.S., Marcellin, M.W.: JPEG 2000: Image Compression Fundamentals, Standards and Practice. Kluwer Academic Publishers, Norwell, MA, USA (2001)
24. Theis, L., Shi, W., Cunningham, A., Huszr, F.: Lossy image compression with compressive autoencoders. In: the 5th Int. Conf. on Learning Representations (2017), http://arxiv.org/abs/1703.00395
25. Toderici, G., Vincent, D., Johnston, N., Hwang, S.J., Minnen, D., Shor, J., Covell, M.: Full resolution image compression with recurrent neural networks. In: IEEE Conf. on Computer Vision and Pattern Recognition (CVPR) (2017). https://doi.org/10.1109/CVPR.2017.577., http://arxiv.org/abs/1608.05148
26. Wang, Z., Simoncelli, E.P., Bovik, A.C.: Multiscale structural similarity for image quality assessment. In: The Thirty-Seventh Asilomar Conference on Signals, Systems & Computers (2003). https://doi.org/10.1109/ACSSC.2003.1292216
27. Zhang, Y., Tian, Y., Kong, Y., Zhong, B., Fu, Y.: Residual dense network for image super-resolution. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2018)
8 Supplementary material

As denoted in the main paper, our JointIQ-Net is the first work that surpasses the coding efficiency performance of the the most recent and advanced image compression coding, VVC Intra Coding (VTM 7.1 [1]), that has been almost finalized for standardization. This Supplemental Material provides a plenty of experimental results that support the superiority of our JointIQ-Net against the state-of-the-art (SOTA) image compression methods. As shown, our JointIQ-Net has yielded a substantial improvement on coding efficiency against the SOTA methods.

8.1 Experimental results on CLIC and Tecnick imagesets

To thoroughly inspect the effectiveness of our JointIQ-Net, we further performed comparison experiments between our model and the SOTA methods including VVC Intra coding (VTM 7.1 [1]), BPG [8], and Lee et al. [18]'s approach, over two different image datasets, the CLIC [2] validation set and the SAMPLING test set (color format: RGB, bit depth: 8 bits per channel, resolution: 1200 × 1200) of Tecnick [4] image set.

Table 3: BD-rate gains of our JointIQ-Net against the VVC Intra coding (VTM 7.1 [1]), BPG [8], and Lee et al. [18]'s approach for CLIC [2] validation set, and against the VTM 7.1 [1], BPG [8], and Lee et al. [18]'s approach for Tecnick [4] image set. The third row shows the coding gains of our MSE-optimized model in terms of PSNR versus BD-rate, and the fourth row indicates the coding gains of our MS-SSIM optimized model in terms of MS-SSIM versus BD-rate.

|                | CLIC [2]       | Tecnick [4]    |
|----------------|----------------|----------------|
|                | VVC Intra [1]  | BPG [8]        | Lee [18]        |
|                | VVC Intra [1]  | BPG [8]        | Lee [18]        |
| MSE opt.       | 4.85%          | 28.16%         | 21.03%          |
|                | 7.12%          | 35.93%         | 28.21%          |
| MS-SSIM opt.   | 52.60%         | 62.19%         | 21.25%          |
|                | 37.85%         | 54.78%         | 21.97%          |

Table 3 shows the BD-rate gains of our JointIQ-Net against the VTM 7.1 [1], BPG [8], and Lee et al. [18]'s approach for CLIC [2] validation set, and against the VTM 7.1 [1], BPG [8], and Lee et al. [18]'s approach for Tecnick [4] image set. It is clear in Table 3 that our JointIQ-Net significantly improves the coding efficiency over the SOTA methods, especially outperforming the VTM 7.1 [1] with average 4.85% and 7.12%, respectively, for the CLIC [2] validation set and the Tecnick [4] image set. It should be noted that we performed one line of padding to each input image (feature map) at a convolution layer when down-scaling is needed. That is, when the number of horizontal (vertical) lines in an input image (feature map) is odd and the input image (feature map) is needed to be down-scaled, one more horizontal (vertical) line is padded in the most bottom (the most right) to have an even line number before down-scaling. Correspondingly, the decoder removes the padded horizontal (vertical) line, at each scale, based on the transmitted original input size. Because the down-scaling and
up-scaling architectures of the encoder and decoder are symmetric, the decoder can distinguish the unnecessary lines by emulating the padding area decision process of the encoder. In our experimental results, furthermore, the sizes of file headers indicating the original input sizes were included in bpp calculation.

Fig. 9 and 10 show the coding efficiency curves for the results in Table 3 for the CLIC [2] validation set and the SAMPLING testset of Tecnick [4] imageset, respectively. It should be noted in Fig. 9 and 10 that our JointIQ-Net outperforms all the SOTA methods over the entire bpp range.
Fig. 9: Rate-distortion curves of our JointIQ-Net and the SOTA methods, VTM 7.1 [1], BPG [8], for the CLIC validation image dataset [2]. The top and bottom plots represent RD-curves in terms of PSNR and MS-SSIM, respectively.
Fig. 10: Rate-distortion curves of our JointIQ-Net and the SOTA methods, VTM 7.1 [1], BPG [8], for the TECNICK image dataset [4]. The top and bottom plots represent RD-curves in terms of PSNR and MS-SSIM, respectively.
8.2 Subjective comparisons of the reconstructed images

Figs. 11 and 12 show the decoded images of KODIM04 and KODIM07 by our MSE-optimized JointIQ-Net, VTM 7.1 [1], Lee et al. [18]'s and BPG [8] in the clockwise order. As shown in Figs. 11 and 12, the decoded images by our JointIQ-Net look visually more pleasing over the other decoded images by the SOTA methods.

Figs. 13 and 14 are the decoded images of KODIM01 and KODIM13 by our MS-SSIM-optimized JointIQ-Net, VTM 7.1 [1], LEE et al. [18]'s and BPG [8] in the clockwise order. As also shown in 13 and 14, our JointIQ-Net yields the decoded images with better perceptual quality over the other decoded images by the SOTA methods. Note that we cropped the both sides of the decoded images to fit the page width in the case of the horizontally long images (KODIM01, KODIM07, and KODIM13) for convenient visualization.
Fig. 11: Subjective comparison of decoded images by our JointIQ-Net, VTM 7.1 [1], Lee et al. [18]'s approach, and BPG [8] in the clockwise order. Top-left, our JointIQ-Net (MSE-optimized; bpp, 0.2035; PSNR, 33.1097); top-right, VTM 7.1 [1] (bpp, 0.2041; PSNR, 32.9663); bottom-left, BPG [8] (bpp, 0.2078; PSNR, 32.0406); bottom-right, Lee et al. [18]'s method (MSE-optimized; bpp, 0.2040; PSNR, 32.2065)
Fig. 12: Subjective comparison of decoded images by our JointIQ-Net, VTM 7.1 [1], Lee et al. [18]'s approach, and BPG [8] in the clockwise order. Top-left, our JointIQ-Net (MSE optimized; bpp, 0.1243; PSNR, 31.3978); top-right, VTM 7.1 [1] (bpp, 0.1248; PSNR, 30.9643); bottom-left, BPG [8] (bpp, 0.1188; PSNR, 29.4102); bottom-right, Lee et al. [18]'s method (MSE optimized; bpp, 0.1043; PSNR, 29.46)
Fig. 13: Subjective comparison of decoded images by our JointIQ-Net, VTM 7.1 [1], Lee et al. [18]’s approach, and BPG [8] in the clockwise order. Top-left, our JointIQ-Net (MS-SSIM optimized; bpp, 0.2004; MS-SSIM, 0.9528); top-right, VTM 7.1 [1] (bpp, 0.1978; MS-SSIM, 0.9278); bottom-left, BPG [8] (bpp, 0.1920; MS-SSIM, 0.9136); bottom-right, Lee et al. [18]’s method (MS-SSIM optimized; bpp, 0.2073; MS-SSIM, 0.9488)
Fig. 14: Subjective comparison of decoded images by our JointIQ-Net, VTM 7.1 [1], Lee et al. [18]'s approach, and BPG [8] in the clockwise order. Top-left, our JointIQ-Net (MS-SSIM optimized; bpp, 0.2442; MS-SSIM, 0.9319); top-right, VTM 7.1 [1] (bpp, 0.2409; MS-SSIM, 0.8719); bottom-left, BPG [8] (bpp, 0.2760; MS-SSIM, 0.8699); bottom-right, Lee et al. [18]'s method (MS-SSIM optimized; bpp, 0.2630; MS-SSIM, 0.9313)