Soft then Hard: Rethinking the Quantization in Neural Image Compression Supplementary Material

Appendix A: Illustrative Task

In our main paper, we conduct an illustrative experiment to show that additive uniform noise is superior in learning an expressive latent space for compression. Here we introduce the detailed settings of this experiment.

The core of this experiment is an image compression task. We build the compression model that first encodes the image from MNIST dataset to latent variables. We then try three quantization methods to discretize the latent variables for end-to-end optimization. A decoder will correspondingly generate the reconstruction from the quantized latent variables. Since the image resolution in MNIST dataset is 28×28, the network architecture is designed as follows:

Table 1. Network architecture in this illustrative task.

Encoder	**Decoder**
Conv: 5×5 c32 s2	Deconv: 7×7 c32 s1
LeakyReLU	LeakyReLU
Conv: 5×5 c32 s2	Deconv: 5×5 c32 s2
LeakyReLU	LeakyReLU
Conv: 7×7 c4 s1	Deconv: 5×5 c1 s2

This model will transform the image to a four-dimension latent vector, *i.e.*, the $28 \times 28 \times 1$ image will be mapped to the $1 \times 1 \times 4$ latent variable. Ideally, one continuous real number is able to represent any information if the transform network is very powerful. However, the encoder network here is not strong enough. We thereby design to restrict the latent capacity to investigate the latent representation ability that is learned with different quantization methods.

The model is optimized for the rate-distortion objective. The distortion is measured by mean square error between the original image $\boldsymbol{x}$ and the reconstructed image $\hat{\boldsymbol{x}}$. The rate here is measured by the $\mathcal{L}_2$ norm of the quantized latent variables as the continuous log-likelihood $\log p(\tilde{\boldsymbol{y}})$. It is equal to assume a zero-mean Gaussian distribution with fixed scale on $\tilde{\boldsymbol{y}}$. The overall loss function is:

$$\begin{aligned}\mathcal{L} &= \mathcal{L}_{rate} + \lambda \cdot \mathcal{L}_{distortion} \\ &= \mathcal{L}_2(\tilde{\boldsymbol{y}}) + \lambda \cdot \mathcal{L}_{MSE}(\hat{\boldsymbol{x}}, \boldsymbol{x}).\end{aligned} \tag{1}$$

We set the Lagrange Multiplier λ as 10 and use Adam optimizer with learning rate 1e-3 for optimization. We visualize the results in our main paper by selecting the best model during the total 80-epoch training process.

Appendix B: Train-Test Mismatch

This section provides evidences along with some analyses to show that the mismatch between training and test phases is more serious in complex compression model.

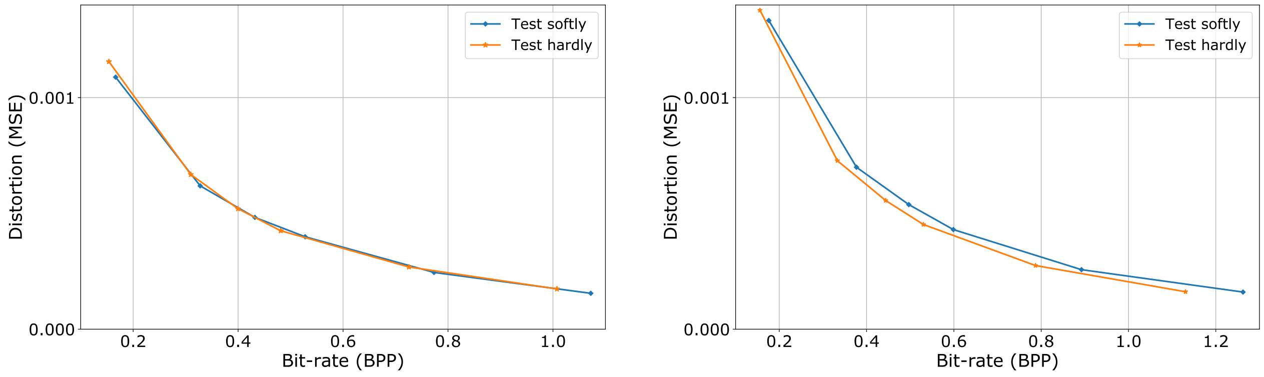
The train-test mismatch is measured by the performance gap between soft quantization (additive uniform noise) and hard rounding. Specifically, we can try to use additive uniform noise to test the (estimated) compression performance on Kodak dataset, although it is not a practical compression process. In Table 2, we present the distortion gap between training and test phases that is measured by the difference between the estimated PSNR value and the true PSNR value: $\mathrm{Gap} = \mathrm{PSNR}_{\mathrm{soft}} - \mathrm{PSNR}_{\mathrm{hard}}$.

Table 2. Distortion mismatch between training and test phases.

λ	192	512	768	1024	2048	4096
Baseline-1 Gap	**0.26**	**0.33**	**0.33**	**0.28**	**0.40**	**0.50**
Baseline-2 Gap	0.14	0.17	0.14	0.21	0.29	0.04

Baseline-1 is the model of (Cheng et al., 2020), more powerful than Baseline-2 (Minnen et al., 2018). We can observe that the distortion gap is more serious in the complex base model (Baseline-1). We deduce that perhaps it is due to the posterior collapse issue, since a sufficiently powerful decoder will tend to ignore the posterior in VAEs.

In addition, we draw both the estimated and the true rate-distortion (RD) curves upon these two base models as shown in Figure 1, *i.e.*, test with soft quantization (additive uniform noise) and test with hard quantization. Here we directly compute the mean square error to stand for distortion. When baseline is a complex model (Cheng et al., 2020), the true rate-distortion curve coincides with the estimated curves as shown in Figure 1a. However, it would be surprising that the true RD performance of Baseline-2 is better than the estimated performance that corresponds to the soft training objective. Actually, it is reasonable because the noise-relaxed compression models are optimized to minimize the variational upper bound of actual rate. Therefore, the estimated rate is larger than the true rate. From another view, in simple



(a)

(b)

Figure 1. The rate-distortion performance mismatch between soft quantization (additive uniform noise) and hard rounding. (a) Base model is (Cheng et al., 2020). (b) Base model is (Minnen et al., 2018). Evaluating on Kodak dataset.

compression models, the actual rate-distortion performance is better than the estimated performance. But this performance improvement (from training to test) is weakened in complex models such as (Cheng et al., 2020), which also implies that the mismatch between training and test phases is more serious in complex compression model.

Appendix C: Ablation Settings

We conduct rigorous ablation study to verify the effectiveness of our proposed techniques, as mentioned in our main paper. We here complement some experimental settings of our ablations and introduce some specific architectures.

Training Details

We train the compression models on the full ImageNet dataset. Original images are cropped to 256×256 patches. Minibatches of 8 of these patches are used to update network parameters that is trained on single RTX 2080 Ti GPU. We apply Adam optimizer with learning rate decay strategy. At the soft training stage, the initial learning rate is 5e-5 and degrades to 1e-5 after 400,000 iterations. We obtain the pre-trained model by selecting the best model during 2,000,000 iterations that is evaluated on Kodak dataset. After accomplishing the soft training stage, we employ scaled uniform noise in the pre-trained model by finetuning the noise-generation branch with 500,000 iterations. Then we conduct ex-post tuning with hard quantization. At this stage, we finetune the decoder for 500,000 iterations as well. During the second and the third stage, the learning rate is 5e-5 and degrades to 1e-5 after 200,000 iterations. The latent channel number is increased at high bitrates to avoid bottleneck issue following (Ballé et al., 2018). Specifically, we assign M=192 channels for low or intermediate bitrates, and assign M=320 channels when λ is 2048 or 4096.

Reproducing Details

We investigate the effects of our methods upon three base models (Minnen et al., 2018; Cheng et al., 2020; Guo et al., 2020). All of them are reproduced by us with Pytorch implementations. The network structures are reproduced as their paper reported exactly. Our reproduced performance has a gap to their reported statistics, which may be caused by the difference of training data. Specifically, we use the full ImageNet training set without extra selection. While (Minnen et al., 2018) do not mentioned the training set, (Cheng et al., 2020) adopt the subset of ImageNet for training with coarse selection and (Guo et al., 2020) use some high-resolution datasets for training. Therefore, there is a gap between our reproduced results and their reported results.

Structures of the New Branch

Our proposed soft-then-hard strategy does not require additional network parameters. Our proposed another technique, the scaled uniform noise, requires a new branch to generate noise scales. Since the value of noise scale is positive, we adopt an exponential layer to activate the final output of this branch. The specific structure of this branch is shown in Figure 2.

Appendix D: More Experimental Results

Here we first provide the zoom-in RD-curves about the ablation study in base model (Cheng et al., 2020) for better visualization. As shown in Figure 3, our proposed scaled uniform noise (SUN) achieves considerable improvements by around 0.1 dB at intermediate bitrates.

As shown in Figure 4, we also present the results of deploying our methods in base model (Guo et al., 2020), which delivers the state-of-the-art image compression performance. It even outperforms the H.266/VVC standard on Kodak dataset (we use the VTM8.0 anchor (VTM, 2020) with

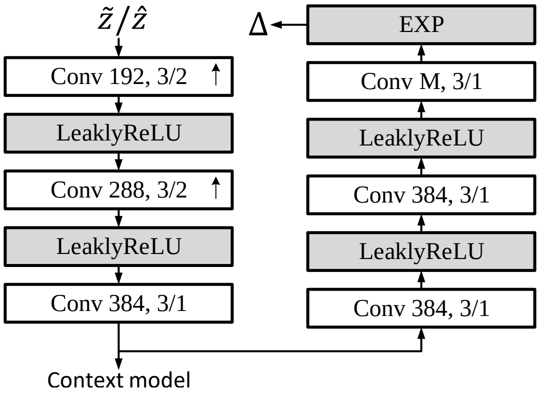


Figure 2. The structure of our proposed branch that generates noise scale Δ. The left several layers are shared with context model. The input is $\tilde{z}$ for soft quantization and $\hat{z}$ for hard quantization. The value of Δ is clamped to avoid extreme value.

YUV444 format and all intra mode). Here the statistics of previous works of neural image compression are taken from their report in papers including (Ballé et al., 2018), (Minnen et al., 2018), (Cheng et al., 2020) and (Guo et al., 2020).

Another important comparison is about the MS-SSIM-optimized case, the metric of which is more consistent with human perceptual quality (Wang et al., 2004). The loss function now is:

$$\begin{aligned}\mathcal{L} &= \mathcal{L}_{rate} + \lambda \cdot \mathcal{L}_{distortion} \\ &= \mathcal{L}_{rate} + \lambda \cdot (1 - \mathcal{L}_{MS-SSIM}(\hat{\boldsymbol{x}}, \boldsymbol{x})).\end{aligned} \tag{2}$$

We train models at four different bitrates with $\lambda = 16, 40, 100, 180$ (latent channel number M=320 when $\lambda =$ 100 or 180). Our methods also bring obvious gains in base model (Cheng et al., 2020) as shown in Figure 5.

In summary, our proposed new methods are robust to bring stable improvements of rate-distortion performance at any bitrate in different base models.

Appendix E: More Visualizations

More reconstruction results are provided here for visual comparisons (Figure 6 and Figure 7). The base model is still (Cheng et al., 2020). And we show both the PSNR-optimized and the MS-SSIM-optimized results.

References

Ballé, J., Minnen, D., Singh, S., Hwang, S. J., and Johnston, N. Variational image compression with a scale hyperprior. *arXiv preprint arXiv:1802.01436*, 2018.

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

Guo, Z., Wu, Y., Feng, R., Zhang, Z., and Chen, Z. 3-d context entropy model for improved practical image compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 116–117, 2020.

Minnen, D., Ballé, J., and Toderici, G. D. Joint autoregressive and hierarchical priors for learned image compression. In *Advances in Neural Information Processing Systems*, pp. 10771–10780, 2018.

VTM. VVC Official Test Model. `https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware_VTM/-/tree/VTM-8.0`, 2020.

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

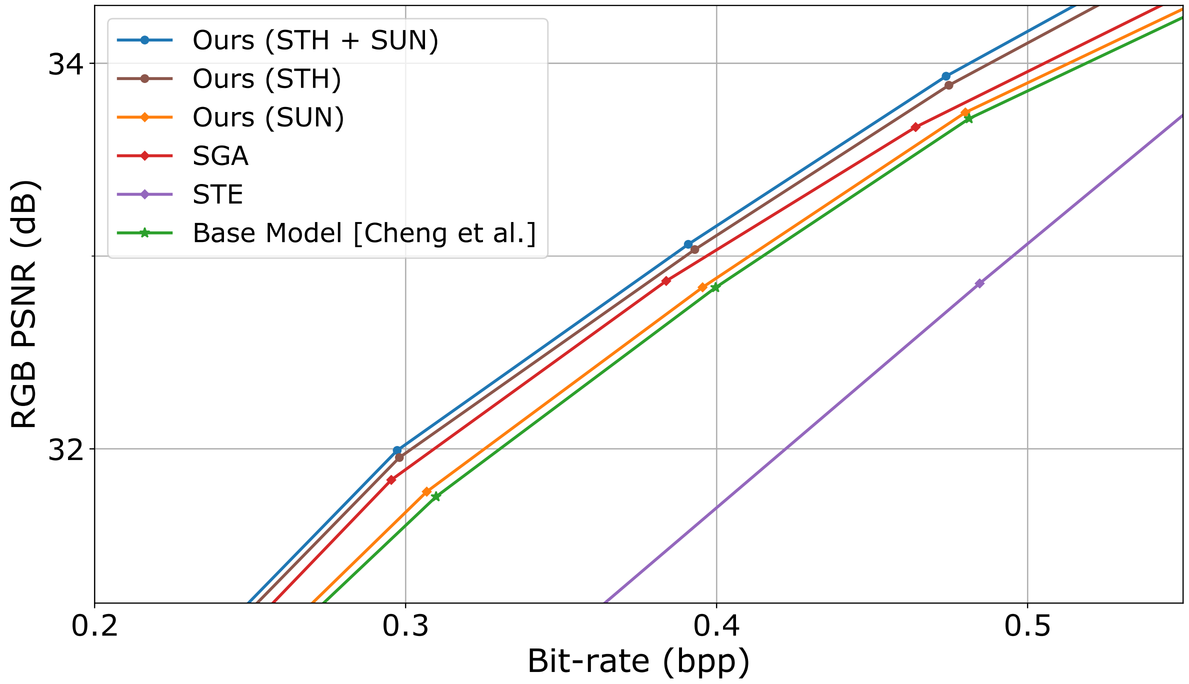


Figure 3. The zoom-in rate-distortion curve in base model (Cheng et al., 2020). Evaluating on Kodak dataset.

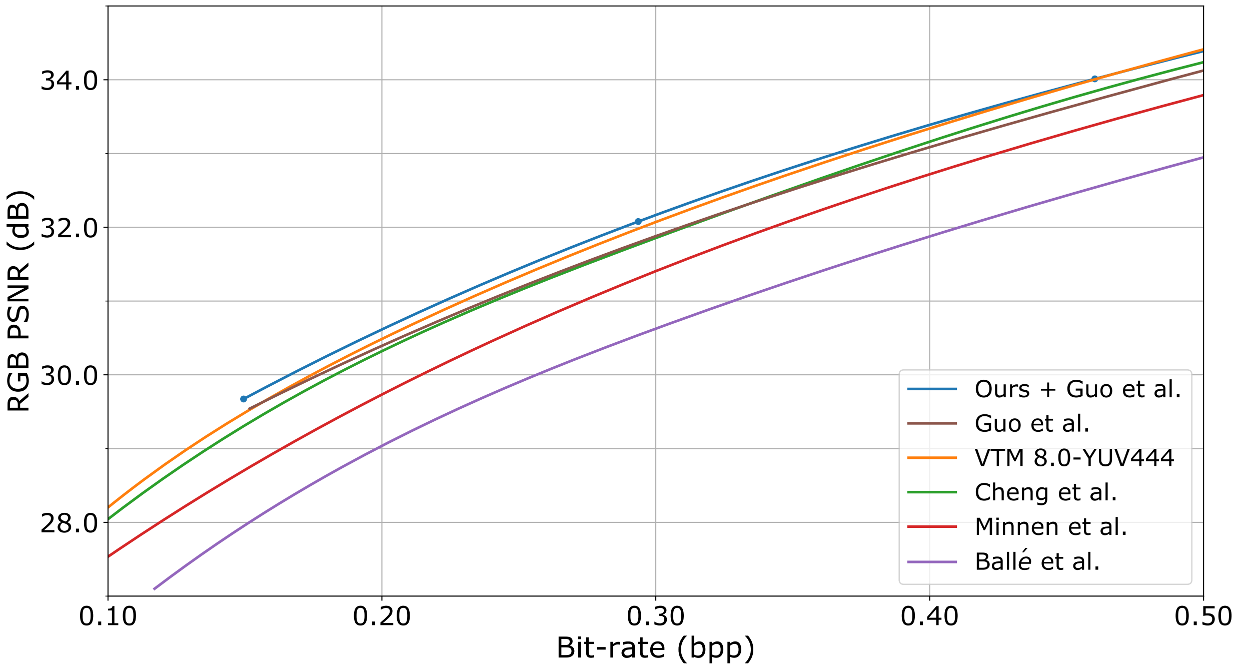


Figure 4. Employing our proposed two techniques in base model (Guo et al., 2020). It helps us achieve the state-of-the-art image compression performance, outperforming all previous neural image compression approaches and the latest image compression standard.

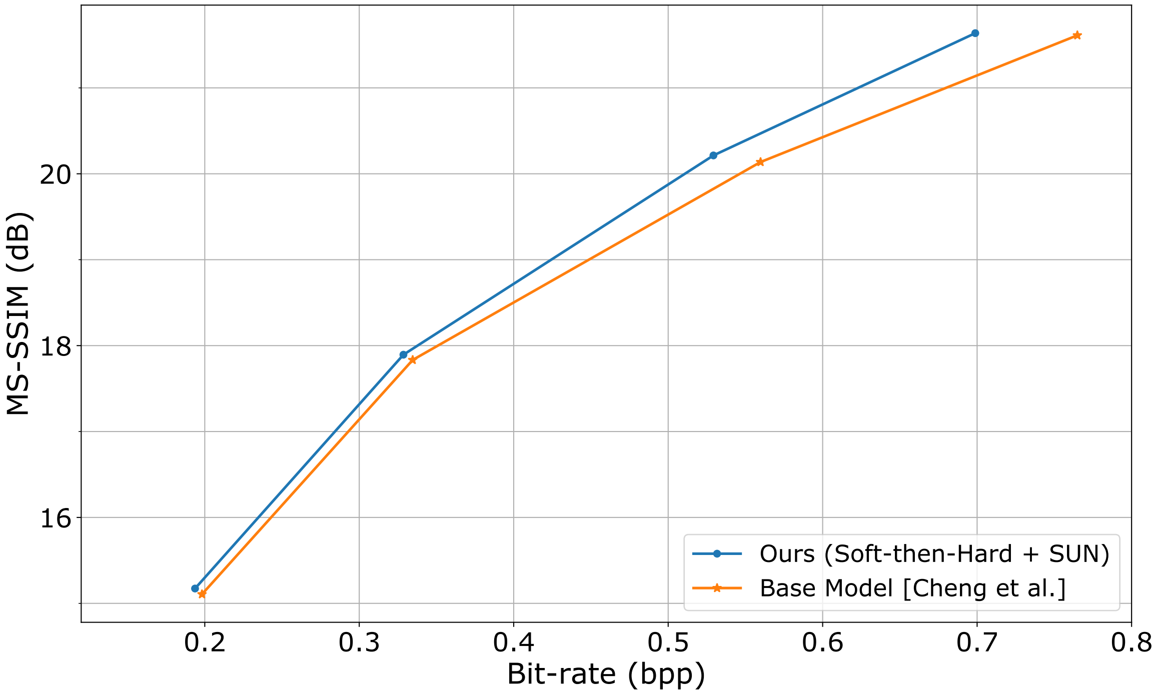


Figure 5. Ablation results in base model (Cheng et al., 2020). Optimized for MS-SSIM.

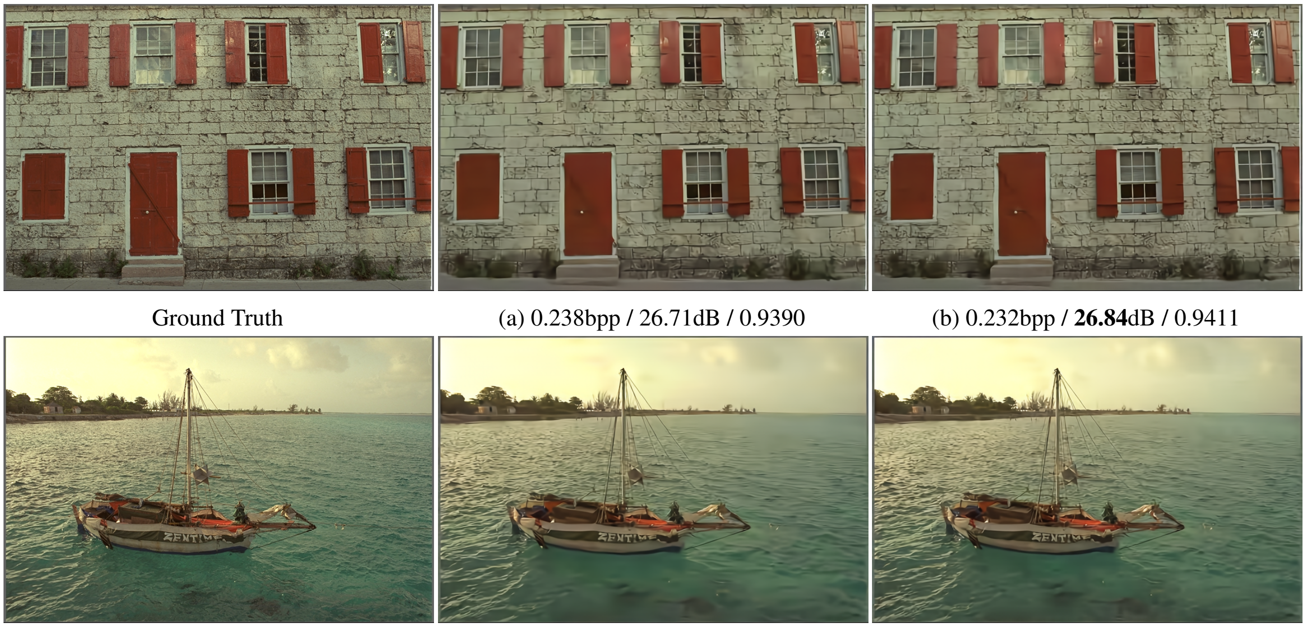
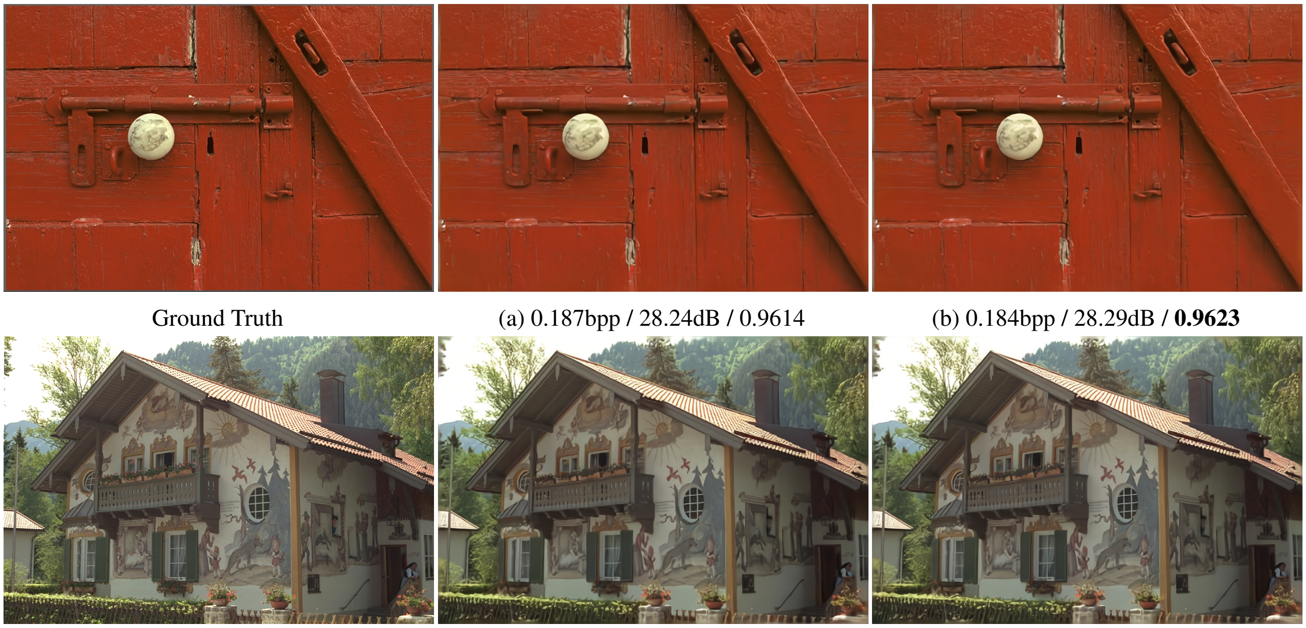

Ground Truth
(a) 0.238bpp / 26.71dB / 0.9390
(b) 0.232bpp / **26.84**dB / 0.9411
Ground Truth
(a) 0.179bpp / 27.98dB / 0.9227
(b) 0.179bpp / **28.17**dB / 0.9269

Figure 6. Visual comparisons. (a) Base model (Cheng et al., 2020) optimized for PSNR. (b) Employing our methods in this base model optimized for PSNR. The statistics are the values of bit-rate (bpp) / PSNR (dB) / MS-SSIM.



Ground Truth

(a) 0.187bpp / 28.24dB / 0.9614

(b) 0.184bpp / 28.29dB / **0.9623**

Ground Truth

(a) 0.238bpp / 24.94dB / 0.9668

(b) 0.234bpp / 24.97dB / **0.9673**

Figure 7. Visual comparisons. (a) Base model (Cheng et al., 2020) optimized for MS-SSIM. (b) Our methods optimized for MS-SSIM.