An End-to-End Deep Generative Network for Low Bitrate Image Coding

Yifei Pei[1], Ying Liu[1], Nam Ling[1], Yongxiong Ren[2], Lingzhi Liu[2]
[1]Department of Computer Science and Engineering, Santa Clara University, Santa Clara, CA, USA
[2]Heterogenous Computing Group, Kwai Inc., Palo Alto, CA, USA

***Abstract*— Generative adversarial network (GAN)-based image compression approaches reconstruct images with highly realistic quality at low bit rates. However, currently there is no published GAN-based image compression approach that utilizes advanced GAN losses, such as the Wasserstein GAN with gradient penalty loss (WGAN-GP), to improve the quality of reconstructed images. Meanwhile, existing deep learning-based image compression approaches require extra convolution layers to estimate and constrain the entropy during training, which makes the network larger and may require extra bits to send information to the decoder. In this paper, we propose a new GAN for image compression with novel discriminator and generator loss functions and a simple entropy estimation approach. Our new loss functions outperform the current GAN loss for low bitrate image compression. Our entropy estimation approach does not require extra convolution layers but still works well to constrain the number of bits during training.**

Keywords— entropy estimation, generative adversarial network, hinge loss, image coding, Wasserstein generative adversarial network, visual communications

I. INTRODUCTION

Deep learning-based image compression approaches have outperformed traditional image compression approaches, such as JPEG2000, in terms of the quality of reconstructed images [1] [2], and are even comparable to BPG [3]. Such approaches use several transform layers trained in an end-to-end manner to minimize the rate-distortion function:

$$min\,(Rate\, +\, \lambda\, Distortion), \tag{1}$$

where λ is a control factor. However, the distortion in these approaches is usually the mean square error (MSE), which alone may not capture the visual reality under a low bit-rate. A least squares generative adversarial network (LS-GAN) was proposed for image compression [4]. Since the adversarial learning nature of GAN can generate photo-realistic images, using a generator as the encoder-decoder for image compression at low bitrates decodes images with better perceptual quality [5]. However, LS-GAN does not control the bit-rate during training. Some other existing GAN-based image compression methods control bit-rates by an entropy loss. The entropy loss is estimated using extra network layers, so the whole GAN model becomes much larger [6], [7], and requires sending side information to decoders.

In this paper, we design an end-to-end GAN-based approach for low bit-rate image compression. Our major contributions are: (1) We propose novel discriminator and generator loss functions to enhance the quality of decoded images. The proposed discriminator loss improves the original Wasserstein GAN with gradient penalty (WGAN-GP) loss with a hinge loss, which only penalizes incorrect classification results. The proposed generator (encoder-decoder) loss includes a content loss that combines mean absolute error (MAE) and multi-scale structural similarity (MS-SSIM), which decodes images with more texture details and higher perceptual quality. (2) We propose a simple entropy estimation method, which produces the entropy loss to trade off the bit rates and decoding quality. This method does not require training additional convolution layers and avoids sending side information to the decoder.

II. BACKGROUND

A. Generative Adversarial Network

A generative adversarial network [8] includes a generator, *G*, to transform noises into photo-realistic images, and a discriminator, *D*, to distinguish generated images from real images. Since the traditional GAN algorithm is unstable in training, the Wasserstein GAN (WGAN) algorithm [9] was proposed to adopt weight clipping to enforce a Lipschitz constraint on the discriminator. Further, the WGAN-GP algorithm [10] adds a soft constraint on the norm of the gradient to encourage the discriminator to be 1-Lipschitz. Its discriminator is trained by minimizing the loss function:

$$L = \mathbb{E}\left[f_D(\tilde{\mathbf{X}})\right] - \mathbb{E}[f_D(\mathbf{X})] + \lambda\mathbb{E}\left[\left(||\nabla_{\hat{\mathbf{X}}} f_D(\hat{\mathbf{X}})||_2 - 1\right)^2\right], \tag{2}$$

where $\hat{\mathbf{X}}$ is the interpolation between the real data $\mathbf{X}$ and the generated data $\tilde{\mathbf{X}}$, and $f_D(\cdot)$ is the discriminator. While $f_D(\mathbf{X})$ is expected to output 1, $f_D(\tilde{\mathbf{X}})$ is expected to output -1. Although these advanced GAN algorithms addressed the problem of instable training, their discriminator loss penalizes both correct and incorrect classification results, which is inefficient.

B. GAN for Deep Image Compression

Agustsson et al. propose an LS-GAN-based approach [11] for extremely low bit-rate image compression [4]. However, their method does not utilize entropy loss to trade off bit rates and image quality, which cannot adapt to different transmission bandwidth requirements. Wang et al. [12] use a multi-scale patch-based discriminator, which can only tell real or fake between local patches, instead of global images. Mentzer et al. propose the HiFiC model, which uses a deep conditional GAN with a hyper-prior for entropy estimation [7]. However, HiFiC does not target low bitrate image compression and adding a hyper-prior model makes GAN model larger. Besides, MSE is adopted in [4] and [7] as the content loss, which is not a perceptual quality metric and is inferior to SSIM in decoding images consistent with the human vision system (HVS) [13]. Furthermore, these GAN models for image compression do not utilize advanced GAN losses; thus, their decoded images may not have satisfying visual qualities.

2023 IEEE International Symposium on Circuits and Systems (ISCAS) | 978-1-6654-5109-3/23/$31.00 ©2023 IEEE | DOI: 10.1109/ISCAS46773.2023.10182028

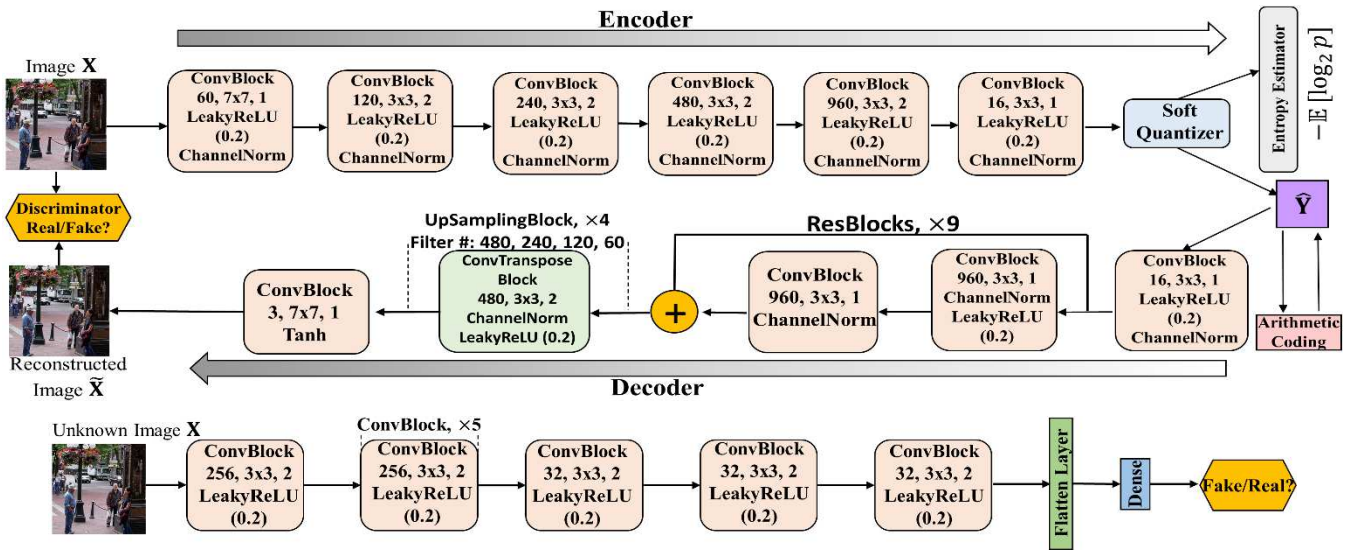

Fig. 1. The proposed hinge GAN-GP architecture with a simple entropy estimator. Top: generator (encoder-decoder); Bottom: discriminator. Numbers in convolution blocks represent the filter number, filter size, and stride, respectively.

III. Proposed Approach

Fig.1 shows our proposed GAN architecture for low bitrate image compression. We propose a novel GAN loss function and simple entropy estimator. The discriminator tells "real" (ground-truth) input images from "fake" (decoded) ones. Our proposed entropy estimator avoids extra model parameters and sending side information to the decoder, which are required by the hyper-prior model [7].

A. Hinge GAN-GP Loss for Image Compression

We improve the WGAN-GP loss in (2), and propose the hinge GAN-GP loss. The discriminator is updated by:

$$\min_D \mathbb{E}\left[max\left(0,1-f_D(\mathbf{X})\right)\right]+\mathbb{E}\left[max(0,1+f_D(\tilde{\mathbf{X}}))\right] + \lambda_1 \mathbb{E}\left[\left(||\nabla_{\hat{\mathbf{X}}} f_D(\hat{\mathbf{X}})||_2-1\right)^2\right], \quad (3)$$

where λ_1 is the weight of the gradient penalty. In (3), $f_D(\mathbf{X})$ is expected to approach 1, and $f_D(\tilde{\mathbf{X}})$ is expected to approach -1. If $f_D(\mathbf{X}) \geq 1$ or $f_D(\tilde{\mathbf{X}}) \leq -1$, the classification is correct and we do not penalize it. Only when $f_D(\mathbf{X}) < 1$ or $f_D(\tilde{\mathbf{X}}) > -1$, the error is penalized, therefore the trained discriminator is more effective than the one trained by (2), which penalizes both correct and incorrect classification results. Besides, we propose to train the generator by:

$$\min_G \ (L_g + \lambda_2 L_{content}), \quad (4)$$

where $L_g = \mathbb{E}[-f_D(\tilde{\mathbf{X}})]$ is the adversarial loss and $f_D(\tilde{\mathbf{X}})$ is expected to output 1,

$$L_{content} = \mathbb{E}[(1-\alpha)\text{MAE}\left(\tilde{\mathbf{X}},\mathbf{X}\right) - \alpha\text{MS-SSIM}(\tilde{\mathbf{X}},\mathbf{X})] \quad (5)$$

is the content loss, and λ_2 controls the contribution of the content loss. The proposed content loss consists of a mean absolute error (MAE) between the decoded image $\tilde{\mathbf{X}}$ and the original image $\mathbf{X}$, which effectively prevents color shift in decoding images, and a negative MS-SSIM score, which captures perceptual loss and decodes images with more texture details.

B. Channel Normalization, Quantization, and Entropy Estimator

Let $\mathbf{Z} \in \mathbb{R}^{C\times W\times H}$ be the output tensor of the activation function of the encoder network's last layer, where C, W, and H are the channel number, width and height of $\mathbf{Z}$, respectively. For each entry $\mathbf{Z}_{ijk}$ in $\mathbf{Z}$, where $1 \leq i \leq C$, $1 \leq j \leq W$ and $1 \leq k \leq H$, the output $\mathbf{Y}_{ijk}$ of the channel normalization [7] is:

$$\mathbf{Y}_{ijk} = \left(\frac{\mathbf{Z}_{ijk}-\mu_{jk}}{\sqrt{\sigma_{jk}^2+\epsilon}}\right)\times\alpha_i+\beta_i, \quad (6)$$

where $\mu_{jk} = \frac{1}{C}\sum_{i=1}^{C}\mathbf{Z}_{ijk}$ and $\sigma_{jk}^2 = \frac{1}{C}\sum_{i=1}^{C}(\mathbf{Z}_{ijk}-\mu_{jk})^2$ are the mean and variance of $\mathbf{Z}$ over channels. ϵ in (6) is a small positive number that prevents the denominator from being zero. α_i and β_i are learnable per-channel offsets. To establish a simple entropy model, we assume $\mathbf{Y}_{ijk}\sim\mathcal{N}(\beta_i,\alpha_i^2)$.

We use the quantization approach proposed in [14]. We adopt a set of integer quantization centers $S=\{-2,-1,0,1,2\}$ and quantize each feature element $\mathbf{Y}_{ijk}$ to its nearest neighbor in set S:

$$\tilde{\mathbf{Y}}_{ijk} = \arg\min_{s_l\in S}|\mathbf{Y}_{ijk}-s_l|. \quad (7)$$

We denote the cumulative distribution function (CDF) of the standard normal distribution as $\Phi(\cdot)$. Since $\mathbf{Y}_{ijk}$ values that lie in the interval $(\tilde{\mathbf{Y}}_{ijk}-0.5, \tilde{\mathbf{Y}}_{ijk}+0.5)$ are quantized as $\tilde{\mathbf{Y}}_{ijk}$, we can approximate the discrete probability $p(\tilde{\mathbf{Y}}_{ijk})$ as $\Phi(\tilde{\mathbf{Y}}_{ijk}+0.5)-\Phi(\tilde{\mathbf{Y}}_{ijk}-0.5)$. Similarly, all $\mathbf{Y}_{ijk}$ values lying in $(1.5,+\infty)$ are quantized as 2, so the discrete probability $p(\tilde{\mathbf{Y}}_{ijk}=2)$ is approximated as $\Phi(+\infty)-\Phi(\tilde{\mathbf{Y}}_{ijk}-0.5)=1-\Phi(\tilde{\mathbf{Y}}_{ijk}-0.5)$. All $\mathbf{Y}_{ijk}$ values lying in

$(-\infty, -1.5)$ are quantized as -2, so $p(\widetilde{\mathbf{Y}}_{ijk} = -2)$ is approximated as $\Phi(\widetilde{\mathbf{Y}}_{ijk} + 0.5) - \Phi(-\infty) = \Phi(\widetilde{\mathbf{Y}}_{ijk} + 0.5)$. Therefore, the estimated entropy of all entries $\widetilde{\mathbf{Y}}_{ijk}$ in $\widetilde{\mathbf{Y}}$ is:

$$L_{entropy} = \mathbb{E}\left[-\log_2 p(\widetilde{\mathbf{Y}}_{ijk})\right] = \begin{cases} \mathbb{E}\left\{-\log_2 \Phi(\widetilde{\mathbf{Y}}_{ijk} + 0.5)\right\}, & \widetilde{\mathbf{Y}}_{ijk} = -2, \\ \mathbb{E}\left\{-\log_2\left[1 - \Phi(\widetilde{\mathbf{Y}}_{ijk} - 0.5)\right]\right\}, & \widetilde{\mathbf{Y}}_{ijk} = 2, \\ \mathbb{E}\left\{-\log_2\left[\Phi(\widetilde{\mathbf{Y}}_{ijk} + 0.5) - \Phi(\widetilde{\mathbf{Y}}_{ijk} - 0.5)\right]\right\}, & \text{otherwise.} \end{cases} \quad (8)$$

In the training process, however, the hard quantization in (7) is not differentiable, so we rely on the differentiable soft quantization:

$$\overline{\mathbf{Y}}_{ijk} = \sum_{s_l \in S} \frac{\exp(-\sigma|\mathbf{Y}_{ijk} - s_l|)}{\sum_{s_h \in S} \exp(-\sigma|\mathbf{Y}_{ijk} - s_h|)} s_l. \quad (9)$$

The quantized feature then becomes:

$$\widehat{\mathbf{Y}}_{ijk} = \text{stop\_gradient}(\widetilde{\mathbf{Y}}_{ijk} - \overline{\mathbf{Y}}_{ijk}) + \overline{\mathbf{Y}}_{ijk}, \quad (10)$$

where the stop_gradient function indicates that in the backward pass, only the gradient of $\overline{\mathbf{Y}}_{ijk}$ is propagated to update network parameters, and the gradient of $\widetilde{\mathbf{Y}}_{ijk} - \overline{\mathbf{Y}}_{ijk}$ is not propagated. In the forward pass, $\widehat{\mathbf{Y}}_{ijk} = \widetilde{\mathbf{Y}}_{ijk}$ as in (7), and we construct a probability dictionary for the five quantized centers. Then, we apply standardized binary arithmetic code to encode all entries $\widehat{\mathbf{Y}}_{ijk}$ in the quantized tensor $\widehat{\mathbf{Y}}$ to binary representations based on the probability dictionary.

C. The Total Generator Loss

We use λ_2 to control the image quality and λ_3 to control the entropy loss so the total loss for our generator in (4) becomes:

$$\min_G L_g^{total} = \min_G \left(L_g + \lambda_2 L_{content} + \lambda_3 L_{entropy}\right), \quad (11)$$

where $L_{entropy}$ is defined in (8).

IV. Experimental Results

We used 235,679 nonoverlapping 256×256 patches extracted from 118,287 images in the COCO dataset [15] to train the GAN models. We used the Adam optimizer with a learning rate of 0.0001 and set the batch size as 24 and the epoch number as 40. We tested the models on all 24 images of 512×768 resolution of Kodak dataset [16] as it is a frequently used test set to evaluate the performance of traditional JPEG, BPG, and learned image compression algorithms [1, 2, 3, 6].

Since we targeted at perceptual quality, we adopted the Fréchet Inception Distance (FID) [17] score and the MS-SSIM as the evaluation metrics for decoded images. The FID score is given by:

$$\text{FID} = ||\boldsymbol{\mu}_{\mathbf{X}'} - \boldsymbol{\mu}_{\widetilde{\mathbf{X}}'}||^2 + \text{Tr}\left(\boldsymbol{\Sigma}_{\mathbf{X}'} + \boldsymbol{\Sigma}_{\widetilde{\mathbf{X}}'} - 2(\boldsymbol{\Sigma}_{\mathbf{X}'}\boldsymbol{\Sigma}_{\widetilde{\mathbf{X}}'})^{1/2}\right), \quad (12)$$

where $\mathbf{X}'$ and $\widetilde{\mathbf{X}}'$ are the features of a pretrained Inception network, extracted from the original image $\mathbf{X}$ and decoded image $\widetilde{\mathbf{X}}$, $\mathbf{X}' \sim \mathcal{N}(\boldsymbol{\mu}_{\mathbf{X}'}, \boldsymbol{\Sigma}_{\mathbf{X}'})$, $\widetilde{\mathbf{X}}' \sim \mathcal{N}(\boldsymbol{\mu}_{\widetilde{\mathbf{X}}'}, \boldsymbol{\Sigma}_{\widetilde{\mathbf{X}}'})$ and $\text{Tr}(\cdot)$ is the trace of a matrix. A lower FID indicates that the distribution of the reconstructed images is closer to the distribution of the original images.

For the GAN models, we set the compressed feature dimensionality to be $16 \times 16 \times 16$. For our proposed model, we empirically set λ_1 in the discriminator loss of equation (3) to be 10, and set λ_2 and λ_3 in the total generator loss of equation (11) to be 100 and 10, respectively, which give good visual qualities of reconstructed images at a low bitrate. We empirically set α in (5) to be 0.84.

Fig. 2 shows the visual qualities of sample decoded test images of BPG, LS-GAN, WGAN-GP, HiFiC, and the proposed hinge GAN-GP. For WGAN-GP, we trained the generator by (11), and trained the discriminator by (2). We observe from Fig. 2 that images decoded by our proposed hinge GAN-GP have the highest visual quality and present clearer texture details. Decoded images of LS-GAN have unfaithful colors, decoded images of both LS-GAN and BPG are blurry in the window area, the girl's nose and red cloth areas, while decoded images of WGAN-GP have obvious noisy artifacts, such as the window, and the green leaves. Decoded images of HiFiC are not smooth on the wall and on the girl's forehead.

In Table 1, we provide the average bpp (bits per pixel), FID and MS-SSIM scores of 24 Kodak test images for all methods compared in Fig. 2. Our proposed hinge GAN-GP has the best MS-SSIM at the lowest bit rate, and its FID is the second best. Although HiFiC achieves the lowest FID score, it requires 65.31% more bit rates than our proposed model. Blurriness and block artifacts in BPG decoded images led to a high FID score, which represents low perceptual quality. This demonstrates that GAN can compress images with higher perceptual quality using adversarial learning.

Table 2 demonstrates the effectiveness of our proposed entropy estimation scheme. As we decrease λ_3 in (11), the average bit rate increases, FID scores decrease and the MS-SSIM values increase. This means our entropy estimator can effectively trade off bit rates and decoding quality. Besides, we observe that the FID of our proposed model at 0.0645 bpp is still better than that of BPG at 0.0948 bpp as shown in Table 1.

To demonstrate the effectiveness of our proposed content loss in (11), we provide an ablation study in Table 3. We fixed the discriminator loss as the proposed hinge GAN-GP loss in (3), and trained our model with three content losses: MSE only, MAE only, and the proposed MAE+MS-SSIM content loss in (5). We tested the three trained models on the 24 Kodak test images, and list the average bpp, MS-SSIM and FID in Table 3. Our GAN model with the proposed MAE+MS-SSIM content loss has the best MS-SSIM and FID at the lowest bpp.

Table 4 shows the parameters of the encoder, decoder, and entropy estimator, as well as the floating-point operations (FLOPs) for the inference of a single 512×768 resolution Kodak image of the GAN models in comparison. Our entropy estimator does not have parameters, therefore, its computational complexity in terms of FLOPs is much lower than that of the hyper-prior entropy estimator used by HiFiC [7]. We do not compare the parameters and FLOPs of the discriminator, since these are not needed at the inference stage, and do not affect the actual encoding and decoding complexity.

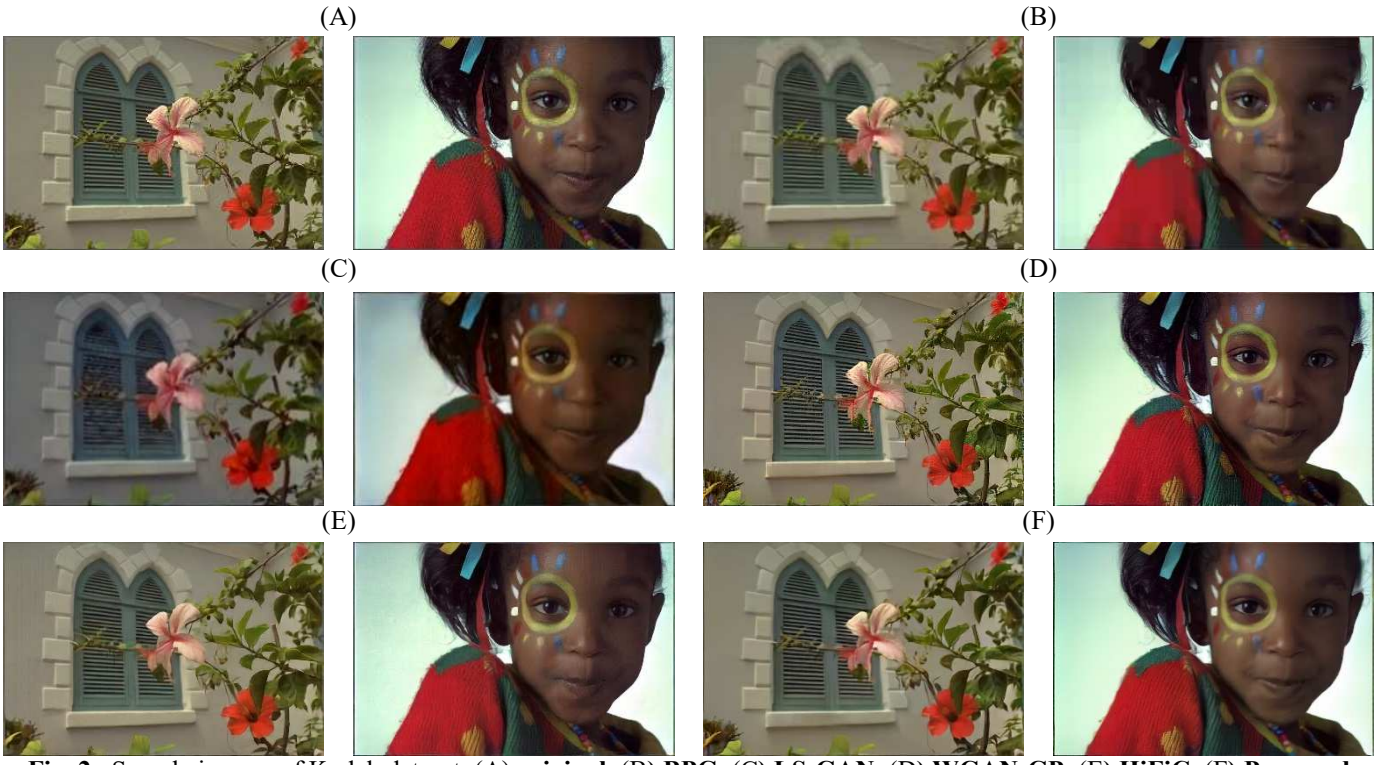

Fig. 2. Sample images of Kodak dataset. (A) **original**; (B) **BPG**; (C) **LS-GAN**; (D) **WGAN-GP**; (E) **HiFiC**; (F) **Proposed hinge GAN-GP**.

Table 1. Comparison of given approaches on 24 Kodak images [16]. The best and the second best value of each metric is marked in red and blue, respectively.

	BPG	LS-GAN	WGAN-GP	HiFiC	Proposed hinge GAN-GP
Average bpp	0.0948	0.1280	**0.0934**	0.1501	**0.0908**
Average MS-SSIM	**0.8932**	0.8401	0.8620	0.8802	**0.8998**
FID	107.40	98.07	70.09	**55.27**	**67.44**

Table 2. Effectiveness of the proposed entropy estimator.

λ_3 in (11)	50	20	10
Average bpp	0.0645	0.0788	0.0908
Average MS-SSIM	0.8672	0.8879	0.8998
FID	97.85	84.80	67.44

Table 3. Our GAN model with different content losses on 24 Kodak images [16].

	MSE	MAE	MAE+MS-SSIM
Average bpp	0.0931	0.0937	0.0908
Average MS-SSIM	0.8719	0.8856	0.8998
FID	96.87	83.57	67.44

Table 4. The number of parameters of GAN models. The FLOPs of each model for the inference of a single Kodak image are included in parentheses.

	Encoder	Decoder	Entropy Estimator
Ours	5, 660, 720 (58.7 G)	155, 011, 683 (518.0 G)	0 (4.67×10^{-4} G)
LS-GAN [4]	5, 660, 688 (58.6 G)	155, 011, 683 (517.0 G)	N/A
HiFiC [7]	5, 626, 116 (58.7 G)	154, 977, 123 (518.0 G)	5, 757, 460 (5.05 G)

V. Conclusion

We propose a hinge GAN-GP with a simple entropy estimator. Our designed entropy estimator without parameters is simple but effective to reduce bit-rates. Our loss functions improve previous GAN losses by retaining more textures and colors at a low bit rate. The images reconstructed by our model are also more natural-looking than those decoded by BPG or other GAN models at lower bit rates. Our future work is to apply our GAN model with the simple entropy estimator to image compressed sensing [18].

REFERENCES

[1] J. Ballé, V.Laparra, and E. P. Simoncelli, "End-to-end optimized image compression," arXiv preprint arXiv:1611.01704, 2016.

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

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

[4] E. Agustsson, M. Tschannen, F. Mentzer, R. Timofte and L. V. Gool, "Generative adversarial networks for extreme learned image compression," Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 221-231.

[5] N. Ling, C.-C. J. Kuo, G. J. Sullivan, D. Xu, S. Lin, H.-H. Huang, W.-H. Peng, J. Liu et al, "The future of video coding," APSIPA Transactions on Signal and Information Processing, vol. 11, no. 1, 2022.

[6] S. Iwai, T. Miyazaki, Y. Sugaya, and S. Omachi, "Fidelity-controllable extreme image compression with generative adversarial networks," 2020 25th International Conference on Pattern Recognition (ICPR). IEEE, 2021, pp. 8235-8242.

[7] F. Mentzer, G. D. Toderici, M. Tschannen, and E. Agustsson, "High-fidelity generative image compression," Advances in Neural Information Processing Systems, vol. 33, pp. 11 913-11 924, 2020.

[8] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," Advances in Neural Information Processing Systems, vol. 27, 2014.

[9] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," International Conference on Machine Learning. PMLR, 2017, pp. 214-223.

[10] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of wasserstein gans," Advances in Neural Information Processing Systems, vol. 30, 2017.

[11] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. Paul Smolley, "Least squares generative adversarial networks," Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 2794-2802.

[12] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, A. Tao, J. Kautz, and B. Catanzaro, "High-resolution image synthesis and semantic manipulation with conditional gans," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 8798-8807.

[13] H. Shi, L. Wang, W. Tang, N. Zheng, G. Hua, "Loss functions for person image generation." British Machine Vision Conference, 2020.

[14] E. Agustsson, F. Mentzer, M. Tschannen, L. Cavigelli, R. Timofte, L. Benini, and L. V. Gool, "Soft-to-hard vector quantization for end-to-end learning compressible representations," Advances in Neural Information Processing Systems, vol. 30, 2017.

[15] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollar, and C. L. Zitnick, "Microsoft coco: Common objects in context," European Conference on Computer Vision. Springer, 2014, pp. 740-755.

[16] R. Franzen, "Kodak lossless true color image suite," source: https://r0k.us/graphics/kodak, vol. 4, no. 2, 1999.

[17] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," Advances in Neural Information Processing Systems, vol.30, 2017.

[18] Y. Pei, Y. Liu, and N. Ling, "Deep learning for block-level compressive video sensing," 2020 IEEE International Symposium on Circuits and Systems (ISCAS). IEEE, 2020, pp. 1-5.