Adversarial Distortion for Learned Video Compression

Vijay Veerabadran[†3], Reza Pourreza[1], Amirhossein Habibian[2], Taco Cohen[2]
[1] Qualcomm AI Research, Qualcomm Technologies, Inc.
[2] Qualcomm AI Research, Qualcomm Technologies Netherlands B.V.
[3] Department of Cognitive Science, University of California San Diego

Abstract

In this paper, we present a novel adversarial lossy video compression model. At extremely low bit-rates, standard video coding schemes suffer from unpleasant reconstruction artifacts such as blocking, ringing etc. Existing learned neural approaches to video compression have achieved reasonable success on reducing the bit-rate for efficient transmission and reduce the impact of artifacts to an extent. However, they still tend to produce blurred results under extreme compression. In this paper, we present a deep adversarial learned video compression model that minimizes an auxiliary adversarial distortion objective. We find this adversarial objective to correlate better with human perceptual quality judgement relative to traditional quality metrics such as MS-SSIM and PSNR. Our experiments using a state-of-the-art learned video compression system demonstrate a reduction of perceptual artifacts and reconstruction of detail lost especially under extremely high compression.

1. Introduction

As the resolution of digitally recorded and streamed videos keeps growing, there is an increasing demand for video compression algorithms that enable fast transmission of videos without loss in Quality-of-Experience. While current video codecs can encode video at low bitrates, this usually results in unpleasant compression artifacts [34, 24]. The application of deep neural networks to develop learned video compression algorithms as explored in recent art [28, 25, 22, 13, 9, 8, 11] produces promising results at solving this issue of perceptual artifacts. However, due to the use of distortion metrics such as MS-SSIM [31] and MSE, the reconstructions tend to be blurry [6]. Generative Adversarial Networks (GANs) have been shown to be capable of producing highly realistic images and videos from random noise inputs [18, 17, 35, 7, 10]. This suggests that the GAN objective more accurately reflects image/video quality as perceived by humans. Indeed the work of [4] has shown that GANs can be used for low-rate high-quality image compression, by augmenting the rate/distortion loss with an adversarial loss. However, so far there is little work on the application of adversarial losses to video compression due to scaling issues.

(a) (b) (c)

Figure 1: Demonstration of the effectiveness of training with adversarial loss. (a) uncompressed frame, learned compression [13] via (b) MS-SSIM distortion, (c) adversarial distortion, at similar bitrates. [See Fig. 5 cation for license information]

We tackle the scaling issue via factorization of our adversarial discriminator into smaller neural network components and show results that demonstrate our compression system's relatively improved perceptual quality even under extreme compression (see Fig. 1). Our model is based on the one proposed by [13], which is a 3D autoencoder with discrete latents and a PixelCNN++[29] prior, trained end-to-end to optimize a rate/distortion loss. Our contributions shall be applicable to other learned video compression systems in general. We also present an ablation study resulting from our search across various formulations of GANs in terms of their architecture and loss functions within the context of lossy video compression. The contributions of this paper are: *i)* we propose adversarial loss to improve the perceptual quality of learned video compression, *ii)* we study techniques to improve the training stability using adversarial loss, *iii)* we study a spatial-temporal factorization of discriminator to enable end-to-end training of deep video compression networks.

[†] Work completed during internship at Qualcomm Technologies, Inc. Qualcomm AI Research is an initiative of Qualcomm Technologies, Inc.

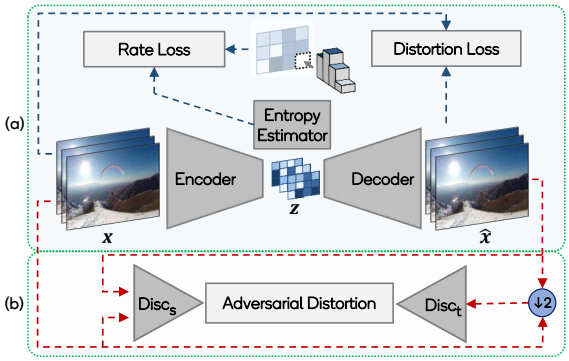

Figure 2: Lossy video compression with adversarial distortion, (a) learned video compression component, (b) adversarial distortion components, where Disc_s and Disc_t represent the spatial and the spatio-temporal discriminators.
[Frames by Ambrose Productions CC BY-SA 3.0 https://creativecommons.org/licenses/by/3.0/legalcode, via YouTube]

2. Learned Video Compression

A learned video compression typically consists of an encoder, decoder, entropy estimator and distortion estimator. All the components are trained end-to-end on a collection of videos $\mathbf{x}$.

Encoder maps an input sequence $\mathbf{x}$ into a latent representation $\mathbf{z} = \mathbf{E}_\phi(\mathbf{x})$. Encoder is a stack of convolutional layers with several down-samplings that reduce the input dimension. In the last layer, encoder employs a quantisation function on the activations to reduce the bit-width of latents $\mathbf{z} = \mathbf{Q}(\tilde{\mathbf{z}})$. Quantizer maps each element, or group of elements, in activations $\tilde{\mathbf{z}}_i$ to a discrete symbol $\mathbf{z}_i \in \{0, \dots, L\}$. Learning discrete representation, as a non-differentiable function, requires adding uniform noise or soft assignment as an approximation. The decoder, which is a stack of convolutional layers with several up-samplings, reconstructs the video given discrete latents $\hat{\mathbf{x}} = \mathbf{D}_\psi(\mathbf{z})$.

Entropy estimator predicts the average number of bits needed to encode latents using a lossless entropy coding schema such as Huffman or arithmetic coding. The bit rate is measured as the cross-entropy between the true distribution of latents $p(\mathbf{z})$ and a density estimated by $P_\theta(\mathbf{z})$ as in Eq. 1. The density estimator $P_\theta(\mathbf{z})$ is parameterized as a neural network usually with an auto-regressive architecture *i.e.* PixelCNN++ [29].

$$\mathbf{H}(\mathbf{z}) = \mathbb{E}_{\mathbf{z}\sim p(\mathbf{z})}[-\log(P_\theta(\mathbf{z}))] \tag{1}$$

Distortion loss measures the difference between the input and reconstructed videos $\mathbf{d}(\mathbf{x}, \hat{\mathbf{x}})$ usually by pixel-wise distances, such as ℓ_1 and ℓ_2, or by more sophisticated metrics such as MS-SSIM. All the aforementioned components are trained end-to-end by minimizing a rate-distortion trade-off as the loss function:

$$\mathbf{d}(\mathbf{x}, \mathbf{D}_\psi(\mathbf{E}_\phi(\mathbf{x})) + \beta\mathbf{H}(\mathbf{z}) \tag{2}$$

	f	g	h
Minimax [12]	$-\log(1+e^{-y})$	$-y-\log(1+e^{-y})$	$-y-\log(1+e^{-y})$
Wasserstein [5]	y	$-y$	$-y$
Least Squares [23]	$-(y-1)^2$	$-y^2$	$(y-1)^2$

Table 1: Component functions for a few adversarial losses.

3. Adversarial Distortion Loss

The distortion, measured by pixel-wise metrics *i.e.* ℓ_1, ℓ_2, and MS-SSIM, are often not perfectly aligned with perceptual quality. Recent work [6] mathematically proves that the distortion and perceptual quality are at odds with each other and minimizing the mean distortion leads to a decrease in perceptual quality. Instead of solely relying on pixel-wise distances, we define the distortion as an adversarial loss between the decoder and a discriminator $\mathbf{C}_\omega$. This setting can be interpreted as training a conditional GAN, where the decoder $\mathbf{D}_\psi$ learns to generate a video given the encoded latents $\mathbf{z}$. The discriminator encourages the decoder to generate videos which reside on the data manifold that improves perceptual quality.

3.1. Stable adversarial training

Adversarial loss Adversarial loss can be defined in various formulations depending on how to specify the component functions f, g and h:

$$\max_{\mathbf{C}_\omega} \mathbb{E}_{\mathbf{x}}[f(\mathbf{C}_\omega(\mathbf{x}))] + \mathbb{E}_{\hat{\mathbf{x}}}[g(\mathbf{C}_\omega(\hat{\mathbf{x}}))] \tag{3}$$

$$\min_{\mathbf{D}_\psi} \mathbb{E}_{\hat{\mathbf{x}}}[h(\mathbf{C}_\omega(\hat{\mathbf{x}}))] \tag{4}$$

Table 1 specifies the component functions for several widely used GAN formulations. We investigate the impact of each formulation on training stability as they have different loss landscapes and gradient behavior; finding the best formulation is hence non-trivial. Minimax loss [12] and Wasserstein loss [5] resulted in fairly decent improvements in terms of reconstruction quality, but we noticed the training to be unstable and time consuming. We also experimented with the Least Squares [23] and Relativistic [16] formulations. Both of these formulations resulted in stable adversarial training. Among these two choices, our best formulation was the Least Squares loss that generated higher quality videos (see section 4.2).

Perceptual loss As a way of further stabilizing our model's adversarial training, we incorporated a semantic loss [21, 30] that minimizes the ℓ_1 of the difference between framewise VGG-19 representations of $\mathbf{x}$ and $\hat{\mathbf{x}}$. This semantic loss resulted in faster and more stable training of our adversarial video compression model.

3.2. Factorized spatial-temporal discriminators

Recent work in training GANs to generate videos points out the advantage of scaling up the training, *i.e.*, using larger batch sizes, deeper models etc. [10]. However, due to scalability issues relating to working with video data and the

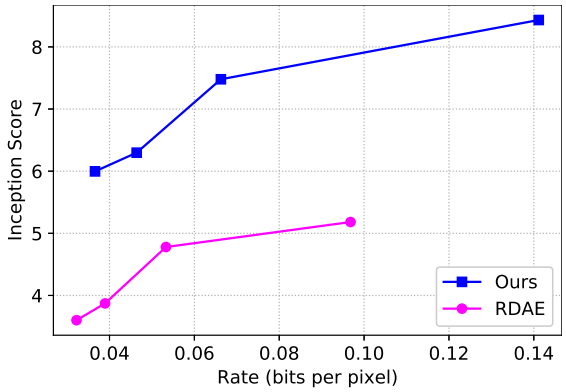

Figure 3: Comparison of RDAE and our method on Kinetics validation set.

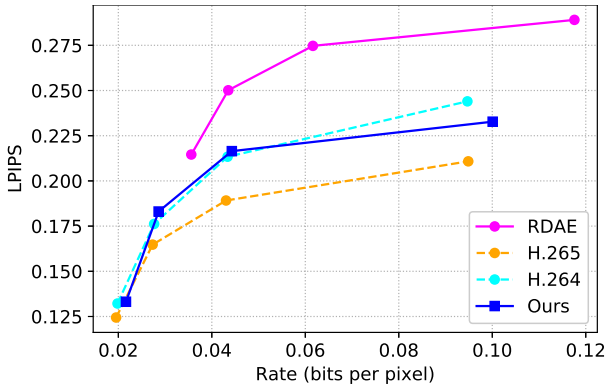

Figure 4: LPIPS comparisons of H.264, H.265, RDAE, and our method on UVG.

models size, we faced difficulty in jointly training all components. In this case, our two choices to scale up our training were: *(i)* finetune the decoder using an adversarial distortion and fixing the prior and encoder, hence loading only the adversarial distortion components in memory, and *(ii)* factorizing our model into smaller components that enable large-scale training. While analyzing the compression performance of the above two choices, we observed that the latter produced higher quality reconstruction at the same bitrates. In order to scale up joint adversarial training for our complete model, we resorted to factorizing the discriminator into two smaller spatial and spatio-temporal discriminators. Both these discriminators were formulated as LSGAN discriminators. The average loss from these discriminators was used to train the decoder.

Putting all together, we train our model end-to-end by optimizing the rate-distortion trade-off Eq. 2 using the following distortion loss:

$$\mathbf{d}(\mathbf{x}, \hat{\mathbf{x}}) = \alpha \left\|\mathbf{x} - \hat{\mathbf{x}}\right\|_2 + \gamma \left\|\sigma(\mathbf{x}) - \sigma(\hat{\mathbf{x}})\right\|_1 + \rho[(\mathbf{C}_\omega(\hat{\mathbf{x}}) - 1)^2] \tag{5}$$

where σ represents the VGG-19 features[1]. These design choices are summarized in our architecture in Fig. 2.

4. Experiments

4.1. Experimental setup

Network architecture We demonstrate the impact of our adversarial training on the Rate-Distortion Autoencoder [13] (RDAE) model which achieves state-of-the-art video compression performance compared to other learned [22, 33] compression methods in terms of MS-SSIM at various bit-rates. Further details on the architecture of RDAE's encoder, decoder, entropy estimator and quantizer can be found in [13]. We employed a 2D ResNet-34 [15] (trained on ImageNet) and a 3D ResNet [14] (trained from scratch) as our spatial and spatio-temporal discriminators respectively. We spatially downsize our spatio-temporal discriminator's input by half in order to save memory, however, we did not temporally subsample our spatial discriminator's input.

Dataset We created a dataset sourced from Kinetics400 [19] by selecting the first 16 frames from each high-quality video and downsampling them to alleviate the existing compression artifacts resulting in a total of 93750 videos for training and 5687 videos for validation. We used random 160×160 crops for training and used full-size frames for validation. We used UVG [2] as the test dataset for comparisons with other methods.

Implementation details We pretrained our video compression model using rate-loss and MSE distortion loss to speed up adversarial training; this step provided a good initialization for adversarially training decoder weights. Our hyperparameter choices for optimizing Eq. 5 are: $\alpha = 0.005$, $\gamma = 0.1$, and $\rho = 0.0001$. We used a batch size of 37, for a total of 12 epochs. We trained our network with 4 values of $\beta = \{0.1, 0.3, 0.5, 0.7\}$ to obtain a rate-distortion curve. All models were trained using Adam optimizer [20] with learning rate of 10^{-4}, $\beta_1 = 0.9$ and $\beta_2 = 0.999$.

4.2. Results

Comparison to state of the art In this section, we compare the performance of our method with a learned video compression method RDAE [13] as well as two non-learned codecs H.265 [27] and H.264 [32]. Fig. 3 shows the comparison of our method with RDAE in terms of Inception Score [26] (IS, ↑) on Kinetics validation set. Incorporation of adversarial training improves IS, and hence the perceptual salience of decoded videos by a great extent.

Due to the small number of videos present in UVG, we are unable to accurately compute IS on this dataset; we report a framewise Learned Perceptual Image Patch Similarity [36] (LPIPS, ↓) score on UVG for this reason. Fig. 4 shows the comparison of our method with RDAE, H.264

[1]We used features from the 4^{th} convolution before the 5^{th} max-pooling layer of an ImageNet-trained VGG-19 network.

(a) Uncompressed frame

(b) H.265, 0.0294 bpp (c) RDAE, 0.0339 bpp (d) Ours, 0.0309 bpp

Figure 5: Visual comparison of H.265, RDAE, and our method at a comparable bitrate. (a) shows an uncompressed frame (frame 49 of Netflix Tango in Netflix El Fuente [3]), (b), (c), and (d) show the close-ups of the reconstructed frame from H.265, RDAE, and our method, respectively. Our method is void of the compression artifacts present in H.265 and over-smoothing present in RDAE under low bit-rates.

and H.265 in terms of LPIPS on UVG dataset. We used `ffmpeg` [1] implementation of H.265 and H.264. From this analysis, we observe that adversarial training improves the perceptual quality of RDAE, resulting in a lower LPIPS score. Our model, albeit underperforms H.265, closes the gap between the learned methods and H.265 in terms of LPIPS to a high extent. In Fig. 5 we compare the visual quality of our lowest rate model with H.265 and RDAE at a similar rate. We can see that our result is relatively free from the compression artifacts usually present in H.265 and RDAE at low bitrate.

Ablation study We trained video compression models separately on each of the 4 GAN loss formulations mentioned in section 3 along with a stabilizing pixel-wise ℓ_1- or ℓ_2-norm of the distance between $\mathbf{x}$ and $\hat{\mathbf{x}}$ (8 GAN models in total, WGAN not reported due to unstable training). We report a summary of these experiments in Table 2 and we decided to use the best performing LSGAN with ℓ_2 pixel-wise loss for our video compression model.

GAN Type	Pixel Loss	MS-SSIM	PSNR (dB)
DCGAN [12]	L1	0.957	26.655
DCGAN	L2	0.958	26.862
RaGAN [16]	L1	0.957	26.554
RaGAN	L2	0.957	26.625
LSGAN [23]	L1	0.96	26.905
LSGAN	L2	**0.961**	**27.032**

Table 2: Ablation study on different GAN losses, PSNR and MS-SSIM Comparisons on Kinetics validation set.

5. Conclusion

In this paper, we have presented a new deep adversarial lossy video compression algorithm that outperforms state-of-the-art learned video compression systems in terms of visual quality. By employing adversarial training for the decoder, we demonstrate reduction in the perceptual artifacts (especially under very low bit-rates) typically present in the reconstructed output. We have also presented an ablation study of our design choices that resulted in our final adversarial compression model.

References

[1] ffmpeg. http://ffmpeg.org/. Accessed: 2020-03-03. 4

[2] Ultra Video Group test sequences. http://ultravideo.cs.tut.fi/. Accessed: 2020-03-03. 3

[3] Xiph.org video test media [derf's collection]. https://media.xiph.org/video/derf/. Accessed: 2020-03-03. 4

[4] Eirikur Agustsson, Michael Tschannen, Fabian Mentzer, Radu Timofte, and Luc Van Gool. Generative adversarial networks for extreme learned image compression. In *ICCV*, 2019. 1

[5] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan. *arXiv*, 2017. 2

[6] Yochai Blau and Tomer Michaeli. The perception-distortion tradeoff. In *CVPR*, 2018. 1, 2

[7] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv*, 2018. 1

[8] Tong Chen, Haojie Liu, Qiu Shen, Tao Yue, Xun Cao, and Zhan Ma. Deepcoder: A deep neural network based video compression. In *VCIP*, 2017. 1

[9] Zhibo Chen, Tianyu He, Xin Jin, and Feng Wu. Learning for video compression. *IEEE Trans. on Circuits and Systems for Video Technology*, 2019. 1

[10] Aidan Clark, Jeff Donahue, and Karen Simonyan. Efficient video generation on complex datasets. *arXiv*, 2019. 1, 2

[11] Abdelaziz Djelouah, Joaquim Campos, Simone Schaub-Meyer, and Christopher Schroers. Neural inter-frame compression for video coding. In *ICCV*, 2019. 1

[12] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NeurIPS*, 2014. 2, 4

[13] Amirhossein Habibian, Ties van Rozendaal, Jakub M Tomczak, and Taco S Cohen. Video compression with rate-distortion autoencoders. In *ICCV*, 2019. 1, 3

[14] Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet? In *CVPR*, 2018. 3

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015. 3

[16] Alexia Jolicoeur-Martineau. The relativistic discriminator: a key element missing from standard gan. *arXiv*, 2018. 2, 4

[17] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv*, 2017. 1

[18] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, 2019. 1

[19] Will Kay, João Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, Mustafa Suleyman, and Andrew Zisserman. The Kinetics Human Action Video Dataset. *arxiv*, 2017. 3

[20] D Kingma and J Ba. Adam: A Method for Stochastic Optimization. 2015. 3

[21] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *CVPR*, 2017. 2

[22] Guo Lu, Wanli Ouyang, Dong Xu, Xiaoyun Zhang, Chunlei Cai, and Zhiyong Gao. Dvc: An end-to-end deep video compression framework. In *CVPR*, 2019. 1, 3

[23] Xudong Mao, Qing Li, Haoran Xie, Raymond YK Lau, Zhen Wang, and Stephen Paul Smolley. Least squares generative adversarial networks. In *ICCV*, 2017. 2, 4

[24] Reza Pourreza, Amir Ghodrati, and Amirhossein Habibian. Recognizing compressed videos: Challenges and promises. In *ICCV Workshops*, 2019. 1

[25] Oren Rippel, Sanjay Nair, Carissa Lew, Steve Branson, Alexander G Anderson, and Lubomir Bourdev. Learned video compression. In *ICCV*, 2019. 1

[26] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, Xi Chen, and Xi Chen. Improved techniques for training gans. In *NeurIPS*. 2016. 3

[27] G J Sullivan, J R Ohm, W J Han, and T Wiegand. Overview of the High Efficiency Video Coding (HEVC) Standard. *IEEE Trans. on Circuits and Systems for Video Technology*, 2012. 3

[28] Michael Tschannen, Eirikur Agustsson, and Mario Lucic. Deep generative models for distribution-preserving lossy compression. In *NeurIPS*, 2018. 1

[29] Aaron Van den Oord, Nal Kalchbrenner, Lasse Espeholt, Oriol Vinyals, Alex Graves, et al. Conditional image generation with pixelcnn decoders. In *NeurIPS*, 2016. 1, 2

[30] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. Esrgan: Enhanced super-resolution generative adversarial networks. In *ECCV*, 2018. 2

[31] Zhou Wang, Alan C Bovik, Hamid R Sheikh, Eero P Simoncelli, et al. Image quality assessment: from error visibility to structural similarity. *IEEE Trans. on Image Processing*, 2004. 1

[32] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra. Overview of the h.264/avc video coding standard. *IEEE Trans. on Circuits and Systems for Video Technology*, 2003. 3

[33] Chao-Yuan Wu, Nayan Singhal, and Philipp Krahenbuhl. Video compression through image interpolation. In *ECCV*, 2018. 3

[34] Kai Zeng, Tiesong Zhao, Abdul Rehman, and Zhou Wang. Characterizing perceptual artifacts in compressed video streams. In *Human Vision and Electronic Imaging XIX*, 2014. 1

[35] Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. Self-attention generative adversarial networks. *arXiv*, 2018. 1

[36] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. *arXiv*, 2018. 3