ADVERSARIAL VIDEO COMPRESSION GUIDED BY SOFT EDGE DETECTION

Sungsoo Kim*† Jin Soo Park*† Christos G. Bampis * Jaeseong Lee *
Mia K. Markey * Alexandros G. Dimakis * Alan C. Bovik*

* The University of Texas at Austin, Austin, TX, USA.
* Netflix Inc., Los Gatos, CA, USA
† Co-primary authors.

ABSTRACT

We propose a video compression framework using conditional Generative Adversarial Networks (GANs). We rely on two encoders: one that deploys a standard video codec and another one which generates low-level *soft edge maps*. For decoding, we use a standard video decoder as well as a decoder that is trained using a conditional GAN. Recent "deep" approaches to video compression require multiple videos to pre-train generative networks that conduct interpolation. By contrast, our scheme trains a generative decoder that requires only a small number of key frames and edge maps taken from a single video, without any interpolation. Experiments on two video datasets demonstrate that the proposed GAN-based compression engine is a promising alternative to traditional video codec approaches that can achieve higher quality reconstructions for very low bitrates.

Index Terms— Video codec, Soft edge detector, conditional Generative Adversarial Networks

1. INTRODUCTION

Video compression is the process of reducing the size of a video file while retaining the perceptual quality of the decompressed data. Developing better video compression techniques lies at the core of applications where more efficient video storage and transmission is essential, such as adaptive video streaming.

Despite the success of traditional video compression standards, there is recent interest for neural network based image and video compression, fueled by the success of deep neural networks (DNNs) in several other image-related applications. DNNs have been applied to image compression [1, 2] and shown to deliver promising performance, especially at low bitrates [3]. Similar ideas have been applied to video compression, e.g., by casting the motion estimation task as an interpolation problem solved by training on a large volume of videos [4, 5, 6]. These approaches have achieved performance approaching that of prevailing traditional codecs such as H.264 and HEVC [5].

Here, we propose a novel video compression framework that uses conditional Generative Adversarial Networks (GANs) guided by low-level maps generated by a newly conceived *soft edge detector*. The detection of substantive changes in edges, or luminance, once a cornerstone of computer vision theory, is regarded as a plausible front-end feature extraction process in biological vision systems [7]. The topic has found little currency in recent years, due to the difficulty and sensitivity to perturbations during compression. Nevertheless, the resilient behavior of DNNs against realistic noise, blur, and other distortions makes the use edge-based approaches for DNN-based compression an intriguing proposition.

While prior DNN-based compression schemes require an interpolation process trained on multiple videos [4, 5, 6], the proposed approach only requires a small number of frames (1%) *from a single video* to train a decoder. An important implication of our work is on re-establishing the importance of edges, as inspired by biological vision, but for informing a modern deep video compression architecture. We also compare the proposed approach to two widely-deployed state-of-the-art video compression standards (H.264 and HEVC) on two public video datasets; KTH [8] and the YouTube pose dataset [9]. A total of 215K frames (131 videos) are used in the comparisons and all of the data are publicly available.

2. RELATED WORK

2.1. Generative adversarial networks

Rather than using random latent codes in GANs [10], conditional GANs include both deterministic and probabilistic traits of latent codes [11] to build a semantic relationship between a latent code and an output code. This approach has been used, for example, to translate pictures from one image space to another, given pairs of images as training data [12].

More recently, GANs have been used for video applications, e.g., for video prediction and generation [13, 14, 15, 16], for estimating pixel motion [17, 18], and for stochastic video prediction [19, 20] and generation [21, 22]. GANs have also been applied to video compression by learning to gener-

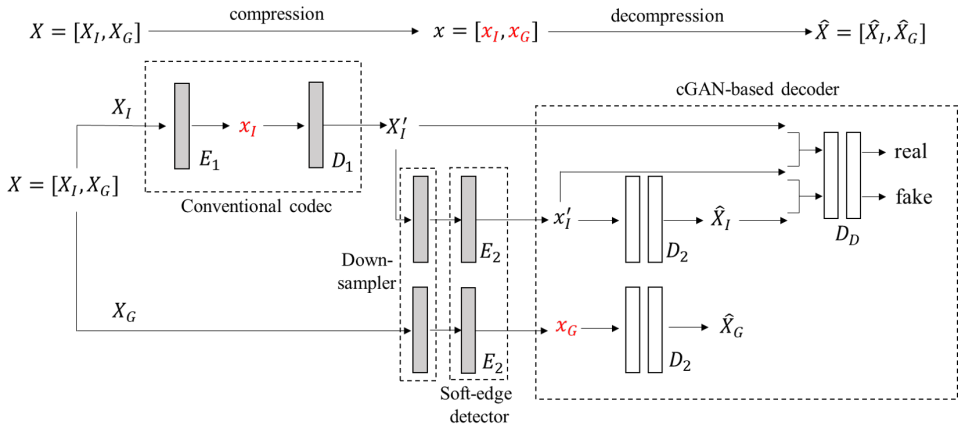

Fig. 1: Proposed framework. A video X is partitioned into two sets of frames: key frames (X_I) and non-key frames (X_G). X_I is compressed using the conventional codec for additional performance gain and deployed consequently to train a decoder D_2 and a discriminator D_D using a newly devised *soft edge detector* E_2. Only D_D and D_2 are trainable (white boxes). Note that a proposed method requires only a small number of key frames (1%) of a single video to train the networks in our experiments.

ate interpolated frames using a large number of videos [4, 5, 6]. While this approach has been effective, it is likely inappropriate for live video streaming, since long term frame predictions can produce severe and systematic artifacts (such as missing or blurred objects). Furthermore, the predictive accuracy of these approaches can be degraded severely in the presence of large or sudden movements of objects. Our approach seeks to combat these problems using soft edge-guided conditional GANs, as explained in Section 3.

2.2. Soft edge detector

While the importance of edge detection has somewhat faded, the zero crossings of bandpass derivative responses remain popular in low-level image analysis tasks [23, 24]. An intriguing aspect of bandpass zero crossings is the plausibility of reconstructing a filtered signal from the zero crossings of multiple responses, motivated by Logan's original theorem [25]. While a number of approaches to image reconstruction from 2D bandpass zero-crossing maps have been proposed [26, 27], the topic has found little currency in recent years, owing to the sensitivity to noise and perturbations of the reconstructions. Nevertheless, because of the highly compact, binary nature of zero-crossing maps, the prospect remains intriguing, particularly in view of the possibility of training deep learning networks to learn zero-crossing based reconstructions on real data, with resilience against realistic noise, blur, and other distortions.

In our approach, frame representations are learned by a conditional GAN guided by soft edge responses. We adapt the terminology of "soft" since our edge detector generates multilevel edge maps, rather than binary ones. The goal of our framework is to reconstruct frames from soft edge maps.

3. PROPOSED FRAMEWORK

We first introduce the notation used throughout the rest of the paper. Let $X \in \mathbb{R}^{W \times H \times 3 \times N}$ denote the set of all frames in a video, having a spatial resolution of $W \times H$ pixels, three (*RGB*) channels and temporal duration of N frames. Also, let $X^{(t)} \in \mathbb{R}^{W \times H \times 3}$ denote the t-th frame of X and $n(\cdot)$ denote the cardinality of a set of frames, i.e., $n(X) = N$. Each channel in a video frame is stored using 8 bits. The small letter x represents the compressed data of X.

We partition the set of video frames X into two subsets: X_I and X_G. X_I is the set of selected key frames and X_G is the set of generated (non-key) frames, which we also refer to as "G-frames". Let $n(X_I) = N_I$ and $n(X_G) = N - N_I$, where $N_I \in \{1, 2, ...N\}$. X_I and X_G can be composed of any *arbitrary* N_I and $N - N_I$ frames from X, respectively. The elements of X_I play a similar role as that of I-frames in conventional codecs and convey similar advantages.

Our GAN-based compression model has two encoding stages (see Fig. 1). At the first stage, the frames in X_I are encoded by some conventional codec (E_1) to generate an encoded representation x_I. Then, the decoder D_1 decodes x_I to reconstruct X'_I. Notably, we can use any conventional codec to implement E_1 and/or D_1. X'_I and X_G are subjected to a second stage of encoding; the soft edge detector (E_2) encodes X'_I and X_G to generate x'_I and x_G.

The second stage encoder, E_2, is composed of a newly-devised *soft edge detector* generating soft edge maps, Q. We apply the Canny edge detector [23, 24] to find an edge pixel map of key frames, $[\mathrm{I}_{i,j}]$, where

$$I_{i,j} = \begin{cases} 1, & \text{if the pixel at } (i,j) \text{ belongs to an edge} \\ 0, & \text{else,} \end{cases}$$

Following edge detection, the soft edge detector extracts color information from the raw frame and performs vector quantization to form clusters of colored pixels which are mapped to edge pixels, i.e.,

$$q'_{i,j} = \mathbb{V}_k \left(X^t \odot I_{i,j} \right) \tag{1}$$

where $\odot$ is the Hadamard product, or entry-wise product, and $\mathbb{V}_k(\cdot)$ is a vector quantizer which uses the k-nearest mean [28], to form $k-1$ clusters of colored pixels mapped to $I_{i,j} = 1$.

To achieve further compression, we apply down-sampling to reduce the spatial resolution of each frame, while the number of frames remains unchanged. Then, we perform run-length encoding [29] and Huffmann encoding [30] for further lossless compression. This encoding process is not as efficient as arithmetic coding, but serves as a proof of concept for our scheme.

The second stage decoder D_2 and the discriminator D_D are trained in an adversarial manner. We utilize the conditional GAN framework of Isola [12] to train the decoder D_2 and the discriminator D_D by parsing the compressed data x'_i into both original data X'_I and decompressed data $\hat{X}_I$ in an adversarial manner. Note that only key frames are used to train D_2 and D_D.

The basic idea behind GANs is a min-max game by two adversarial networks. The objective function of the min-max game can be expressed as

$$\begin{aligned} L\left(D_2, D_D\right) =& \mathbb{E}_{x'_I, X'_I}\left[\log(D_D\left(x'_I, X'_I\right))\right] \\ &+ \mathbb{E}_{x'_I}\left[\log\left(1 - D_D\left(x'_I, D_2\left(x'_I\right)\right)\right)\right] \end{aligned} \tag{2}$$

where

$$\begin{aligned} X'_I &= D_1(E_1(X_I)) \\ x'_I &= E_2(f(X_I)). \end{aligned} \tag{3}$$

Naturally, the encoder E_2 cannot learn evolving representations of non-key frames using information from key frames only. Therefore, we employ a conditional GAN to train D_2 using pairs of key frames X'_I decompressed by some conventional codec and their corresponding soft edge maps x'_I. During training, D_2 learns associations between key frames and the corresponding soft edge maps. Once D_2 is trained, it can be used by the soft edge maps x_G to reconstruct the G-frames X_G (non-key frames), which the decoder has never seen before. When the soft edge map contains more information, it can guide the decoder to reconstruct frames with better quality (see section 4.2).

4. EXPERIMENTAL RESULTS

4.1. Experimental evaluation details

To conduct a quantitative performance analysis, several video quality assessment (VQA) algorithms were deployed: PSNR, SSIM [31], MS-SSIM [32] and VMAF [33]. For our experiments, the following two publicly available video datasets were used:

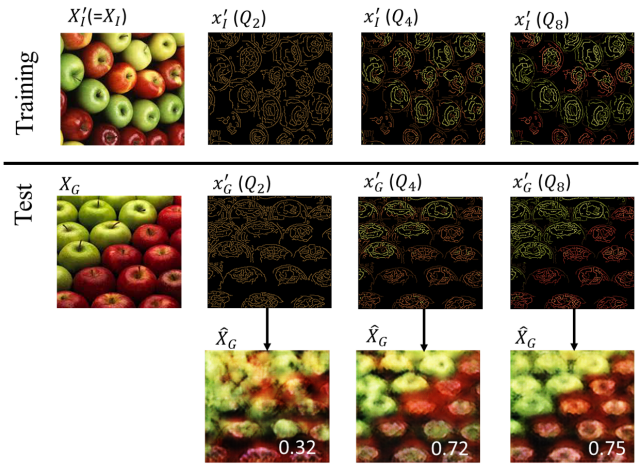

Fig. 2: Performance of the proposed framework for different quantization levels, k, of the *soft edge detector* (Q_k). As k increases, the reconstructed representations become more similar to an original frame.

KTH dataset [8]: The KTH dataset includes 600 short videos (6 types of actions, 25 people, 4 different scenarios). Each video contains between 400 and 800 frames. From these, we randomly collected 100 videos.

YouTube Pose dataset [9]: The Youtube pose dataset is comprised of 45 YouTube videos. Each video contains between 2,000 and 20,000 frames. We excluded 13 of the videos owing to their long durations (>15 min; 7 videos), frequent scene changes (6 videos), and one duplicated video. The remaining 31 videos were included in our analysis.

We compared the performance of our conditional GAN-based compresssor with H.264 (see [34] for a comparison with H.265). We do not compare our scheme with recent video compression models based on DNNs [4, 5, 6], since (1) our model is only trained on a subset of frames from a single target video without any pre-training and (2) our model does not rely on any interpolation.

In our experiments, the H.264 encoder and decoder was used for E_1 and D_1. The original frames were re-sized and cropped to size $2^8 \times 2^8$ and a stride size of 2×2 was used over eight consequent layers for D_2 and D_D. The batch size and the number of epochs were fixed as 1 and 1000, respectively.

4.2. Quality of reconstructed frames according to quantization level of *soft edge detector*

We also examined the relationship between the quality of reconstructed frames and the level of quantization delivered by the soft edge decoder. As a toy example, we applied $k = 2$ quantization to generate a single frame of x'_I. Then, D_2 was

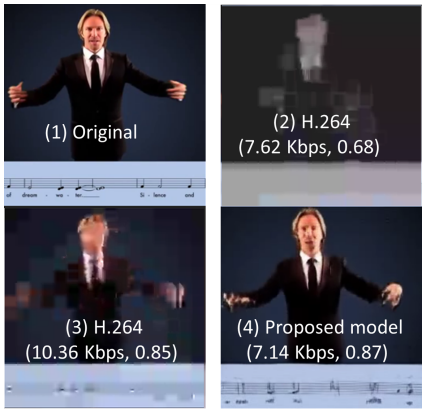

Fig. 3: An original frame (1) and examples of reconstructed frames using H.264 ((2) and (3)) and our model (4). The video bitrate and MS-SSIM score are also reported for each frame. Our approach achieves best visual quality (MS-SSIM of 0.87) at a lower bitrate (7.14 Kbps). Please see https://youtu.be/VHk7G5V6iBs for additional results.

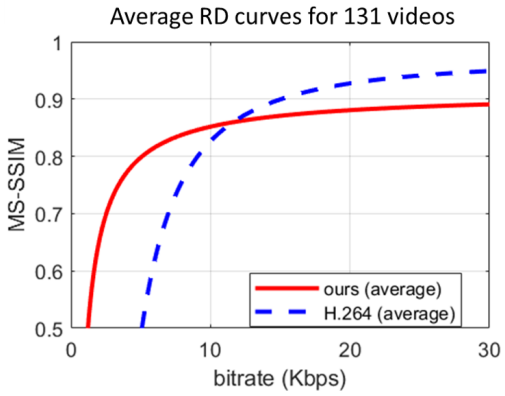

Fig. 4: Average rate-distortion (RD) curves (using MS-SSIM) for 131 videos. In the very low bitrate region (below 10 Kbps), our scheme (red solid curve) yields higher MS-SSIM scores compared to H.264 (blue dashed curve) yielded (see [34] for a similar result for H.265).

trained on a single pair X'_I and x'_I. The trained D_2 was then applied on x'_G to reconstruct $\hat{X}_G$ (Fig. 2). We also repeated this experiment with $k = 4$ and $k = 8$. As the quantization level k was increased, the quality of the reconstructed frames improved qualitatively ($\hat{X}_G$ in Fig. 2). This improvement was also measured by MS-SSIM which increased from 0.32 to 0.72 and 0.75 for k 2, 4, and 8 respectively.

4.3. Comparison with H.264

We compared the proposed approach with H.264 in Fig. 3, which shows the original frame and the corresponding reconstructed frames from H.264 and the proposed approach. The proposed approach achieves higher MS-SSIM score below 10 Kbps (0.87 for 7.14 Kbps) than H.264 (0.68 for 7.62 Kbps).

We also carried out experiments on 131 videos coming from the KTH dataset [8] and the Youtube dataset [9]. Figure 4 plots average RD curves (using MS-SSIM). Notably, for bitrates below 10 Kbps, the proposed scheme achieves significantly higher MS-SSIM scores than H.264. Similar results were observed for other VQA score metrics including PSNR, SSIM and VMAF (see [34] for more details). For larger bitrates, we have found that our model delivers promising results that can be further improved, as we discuss next.

5. LIMITATIONS

There are several limitations in this work which could be addressed. First, our codec delivered worse performance compared to H.264 at bitrates higher than 10 Kbps. This gap might be reduced by larger (deeper) network [35] than the very moderate one used here. Second, we manually select $n(X_I)$, the key frames and a quantization level of the soft edge detector. Exhaustive search or a suboptimal greedy algorithm may further improve the performance at the expense of computational complexity. Third, the 256×256 encoding resolution used in our experiment is low compared to 1080 and even 4K bitrate ladders. Nevertheless, our goal in this paper was to demonstrate a proof of concept of our GAN-based model. Lastly, we observed a flicker problem in some of our reconstructed videos. Along with reduction of temporal redundancy within video, flickering effect needs to be suppressed in further works.

6. CONCLUSIONS

We proposed a video compression framework that is based on conditional GANs guided by a soft edge detection scheme. We demonstrated that the proposed approach can achieve better visual quality than current standard video codecs at very low bitrates and promising performance for higher bitrates. Meanwhile, a nice implication of our approach is that it re-establishes the importance of edges for modern DNN-based video compression architectures. Future work could be targeted at deeper neural network architectures and refining the key frame and quantization level selection steps.

7. ACKNOWLEDGEMENT

This research has been supported by NSF Grants DMS 1723052, CCF 1763702, and AF 1901292.

8. REFERENCES

[1] L. Theis, W. Shi, A. Cunningham, and F. Huszár, "Lossy image compression with compressive autoencoders," *arXiv preprint arXiv:1703.00395*, 2017.

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

[3] E. Agustsson, M. Tschannen, F. Mentzer, R. Timofte, and L. Van Gool, "Generative adversarial networks for extreme learned image compression," *arXiv preprint arXiv:1804.02958*, 2018.

[4] S. Santurkar, D. Budden, and N. Shavit, "Generative compression," in *Picture Coding Symposium (PCS)*, pp. 258–262, 2018.

[5] C.-Y. Wu, N. Singhal, and P. Krähenbühl, "Video compression through image interpolation," *arXiv preprint arXiv:1804.06919*, 2018.

[6] S. Wen, W. Liu, Y. Yang, T. Huang, and Z. Zeng, "Generating realistic videos from keyframes with concatenated gans," *IEEE Trans Circ Syst Video Technol.*, 2018.

[7] D. Marr, "Vision: A computational approach," 1982.

[8] C. Schuldt, I. Laptev, and B. Caputo, "Recognizing human actions: a local svm approach," vol. 3, pp. 32–36, 2004.

[9] J. Charles, T. Pfister, D. Magee, D. Hogg, and A. Zisserman, "Personalizing human video pose estimation," 2016.

[10] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," pp. 2672–2680, 2014.

[11] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv preprint arXiv:1411.1784*, 2014.

[12] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," *arXiv preprint*, 2017.

[13] M. Mathieu, C. Couprie, and Y. LeCun, "Deep multi-scale video prediction beyond mean square error," *arXiv preprint arXiv:1511.05440*, 2015.

[14] C. Vondrick, H. Pirsiavash, and A. Torralba, "Generating videos with scene dynamics," *Advances Neural Info Process Syst.*, pp. 613–621, 2016.

[15] W. Lotter, G. Kreiman, and D. Cox, "Deep predictive coding networks for video prediction and unsupervised learning," *arXiv preprint arXiv:1605.08104*, 2016.

[16] J. Oh, X. Guo, H. Lee, R. L. Lewis, and S. Singh, "Action-conditional video prediction using deep networks in atari games," *Advances Neural Info Process Syst.*, pp. 2863–2871, 2015.

[17] C. Finn, I. Goodfellow, and S. Levine, "Unsupervised learning for physical interaction through video prediction," *Neural Info Process Syst*, pp. 64–72, 2016.

[18] Z. Liu, R. A. Yeh, X. Tang, Y. Liu, and A. Agarwala, "Video frame synthesis using deep voxel flow.," *IEEE Int'l Conf Comput Vision*, pp. 4473–4481, 2017.

[19] A. X. Lee, R. Zhang, F. Ebert, P. Abbeel, C. Finn, and S. Levine, "Stochastic adversarial video prediction," *arXiv preprint arXiv:1804.01523*, 2018.

[20] E. Denton and R. Fergus, "Stochastic video generation with a learned prior," *arXiv preprint arXiv:1802.07687*, 2018.

[21] X. Liang, L. Lee, W. Dai, and E. P. Xing, "Dual motion gan for future-flow embedded video prediction," *IEEE Int'l Conf Comput Vision*, vol. 1, 2017.

[22] W. Xiong, W. Luo, L. Ma, W. Liu, and J. Luo, "Learning to generate time-lapse videos using multi-stage dynamic generative adversarial networks," *IEEE Conf Comput Vision Pattern Recogn.*, pp. 2364–2373, 2018.

[23] D. Marr and E. Hildreth, "Theory of edge detection," *Proc. R. Soc. Lond. B*, vol. 207, no. 1167, pp. 187–217, 1980.

[24] J. Canny, "A computational approach to edge detection," *IEEE Trans Pattern Analysis Machine Intell.*, no. 6, pp. 679–698, 1986.

[25] B. F. Logan Jr, "Information in the zero crossings of bandpass signals," *Bell System Technical Journal*, vol. 56, no. 4, pp. 487–510, 1977.

[26] A. L. Yuille and T. Poggio, "Fingerprints theorems for zero crossings," *JOSA A*, vol. 2, no. 5, pp. 683–692, 1985.

[27] S. Mallat, "Zero-crossings of a wavelet transform," *IEEE Transactions on Information theory*, vol. 37, no. 4, pp. 1019–1033, 1991.

[28] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Trans Info Theory*, vol. 13, no. 1, pp. 21–27, 1967.

[29] A. Robinson and C. Cherry, "Results of a prototype television bandwidth compression scheme," *Proc IEEE*, vol. 55, no. 3, pp. 356–364, 1967.

[30] D. A. Huffman, "A method for the construction of minimum-redundancy codes," *Proc. IRE*, vol. 40, no. 9, pp. 1098–1101, 1952.

[31] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Trans Image Process*, vol. 13, no. 4, pp. 600–612, 2004.

[32] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multiscale structural similarity for image quality assessment," in *Asilomar Conf Signals, Syst Comput, 2003*, vol. 2, pp. 1398–1402, Ieee, 2003.

[33] N. T. Blog, "https://medium.com/netflix-techblog/toward-a-practical-perceptual-video-quality-metric-653f208b9652," 2016.

[34] S. Kim, J. S. Park, C. G. Bampis, J. Lee, M. K. Markey, A. G. Dimakis, and A. C. Bovik, "Adversarial video compression guided by soft edge detection," *arXiv preprint arXiv:1811.10673*, 2018.

[35] 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," *arXiv preprint arXiv:1711.11585*, 2017.