Fast Video Quality Enhancement using GANs

Leonardo Galteri, Lorenzo Seidenari, Marco Bertini, Tiberio Uricchio, Alberto Del Bimbo
[name.surname]@unifi.it
Università degli Studi di Firenze - MICC
Firenze, Italy

ABSTRACT

Video compression algorithms result in a reduction of image quality, because of their lossy approach to reduce the required bandwidth. This affects commercial streaming services such as Netflix, or Amazon Prime Video, but affects also video conferencing and video surveillance systems. In all these cases it is possible to improve the video quality, both for human view and for automatic video analysis, without changing the compression pipeline, through a post-processing that eliminates the visual artifacts created by the compression algorithms.

Generative Adversarial Networks have obtained extremely high quality results in image enhancement tasks; however, to obtain such results large generators are usually employed, resulting in high computational costs and processing time. In this work we present an architecture that can be used to reduce the computational cost and that has been implemented on mobile devices. A possible application is to improve video conferencing, or live streaming. In these cases there is no original uncompressed video stream available. Therefore, we report results using no-reference video quality metric showing high naturalness and quality even for efficient networks.

CCS CONCEPTS

• **Computing methodologies** → **Computer vision tasks**; **Image representations**; **Neural networks**; *Image compression*; • **Information systems** → *Multimedia streaming*.

KEYWORDS

Video quality enhancement, video streaming, video compression, GANs, real-time enhancement.

ACM Reference Format:
Leonardo Galteri, Lorenzo Seidenari, Marco Bertini, Tiberio Uricchio, Alberto Del Bimbo. 2019. Fast Video Quality Enhancement using GANs. In *Proceedings of the 27th ACM International Conference on Multimedia (MM '19), October 21–25, 2019, Nice, France.* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/3343031.3350592

1 INTRODUCTION AND PREVIOUS WORKS

Every day a huge number of videos are created, shared and streamed on the web, within video conferencing and in video surveillance systems. It is necessary to compress these video streams, to reduce the required bandwidth and storage.

The effect of the lossy algorithms typically used is a loss of content fidelity with respect to the original visual data to various degrees of magnitude. An approach to improve the perceived video quality, while maintaining a high compression rate, is to perform filtering on the reconstructed frames, to reduce the effect of the various artifacts. For example, the most recent codecs, such as H.265 and AV1 envisage standardized deblocking filtering.

Improving image quality is a topic that has been thoroughly studied, especially in the case of compression artifact removal. Many approaches are based on image processing techniques [4, 6, 10, 13, 14, 20, 21, 23, 24]. Recently, several learning based methods have been proposed [2, 5, 7, 8, 11, 15, 18, 19, 22], using Deep Convolutional Neural Networks (DCNN), trained to restore image quality; the most recent works [2, 8, 22] use increasingly deep architectures, often employing residual blocks. In [8] we have proposed to use a GAN ensemble and a quality predictor that allows them to restore images of unknown quality.

In this work, we propose a solution to artifact removal based on CNNs trained on large sets of frame patches compressed with different quality factors. Our approach is independent with respect to the compression algorithm used to encode a video; it can be used as a post-processing step on decompressed frames and therefore it can be applied on many lossy compression algorithms such as WebM, AV1, H.264/AVC, and H.265/HEVC. This allows avoiding any modification to the existing compression pipelines, that are often optimized e.g. using dedicated hardware such as GPUs or SoCs. Another advantage is that it can be used with dynamic adaptive streaming approaches, (e.g. DASH), where streams are encoded at different bit rates (and thus at different qualities).

A typical use case in which a high compression is desirable is that of video conferencing, in which video streams must be kept small to reduce communication latency and thus improve user experience. Also in the case of entertainment video streaming, there is a need to reduce as much as possible the required bandwidth, to reduce network congestion and operational costs.

2 METHODOLOGY

We apply adversarial training [9], that recently has shown remarkable performances in image processing and image generation tasks [3, 8, 12], optimizing two networks: a *generator* (G) and a *discriminator* (D), where the generator is fed some noisy input and has the goal to create realistic images (restored) that can misguide the discriminator. On the other hand, the discriminator optimizes a classification loss rewarding solutions that correctly distinguish generated images (restored) from real ones (uncompressed). We consider a frame from a compressed video as an image that has been distorted by some known process. Our goal is to learn some function $G(\cdot)$ (i.e. the generator) able to invert the compression process so that restored images are more similar to uncompressed ones.

MM '19, October 21–25, 2019, Nice, France

ACM ISBN 978-1-4503-6889-6/19/10.
https://doi.org/10.1145/3343031.3350592

Model	# Filters	# Blocks	# Params
GAN baseline [8]	64	16	5.1M
Fast	32	12	1.8M
Very Fast	8	16	145k

Table 1: Parameters of the different GANs used. Compared to the previous work [8], the new "Fast" and "Very Fast" networks have a much smaller number of parameters, resulting in a reduced computation time.

We define the function $G(\cdot)$ as a fully convolutional neural network so to avoiding to have to stick to a precise input resolution for frames and most importantly to allow us to train the network over smaller frame crops and larger batches, speeding up the training. Considering the fact that the noise process induced by compression is local, our strategy does not compromise performance. The weights are learned using a Generative Adversarial Framework.

Generative Network. The architecture of our generator is based on MobileNetV2 [17], which is a very efficient network designed for mobile devices to perform classification tasks. Differently from [8], we replace standard residual blocks with bottleneck depth-separable convolutions blocks to reduce the overall amount of parameters.

After a first standard convolutional layer, feature maps are halved twice with strided convolutions and then we apply a chain of B bottleneck residual blocks. The number of convolution filters doubles each time the feature map dimensions are halved. We use two combinations of nearest-neighbor up-sampling and standard convolution layer to restore the original dimensions of feature maps. Finally, we generate the RGB image with a 1×1 convolution followed by a *tanh* activation, to keep the output values between the $[-1, 1]$ range. In all our trained models we employed Batch Normalization to stabilize the training process. Table 1 reports the number of filters, blocks, and weights of the GAN used in our previous work [8], and two variations of the proposed network, called "Fast" and "Very Fast" since they are designed to attain real-time performance. It can be observed that the new GAN architectures have a much smaller number of parameters, resulting in reduced computational costs, that allow reaching the required real-time performance.

Discriminative Network. This network comprises mostly convolutional layers followed by LeakyReLU activation, with a final dense layer and a sigmoid activation. Since the complexity of this network does not affect the execution time during the test phase, we have chosen for all our experiments a discriminator with a very large number of parameters, thus increasing its ability to discriminate fake patches from real ones. As in [7, 8], sub-patches are fed to this network rather than whole images, because image compression operates at the sub-patch level and those artifacts we aim to remove are generated inside them.

3 THE SYSTEM

All models are trained on the DIV2K dataset [1], that comprises 800 high resolution uncompressed images, which we compress using H.264 to generate degraded frames. As an augmentation strategy, considering the small size of DIV2K, we resize images at 256, 384 and 512 on their shorter side and then we randomly crop a patch of 224×224 pixels with random mirror flipping. Tests on the following Derf collection[1] videos: *Mobile Calendar*, *Park Run*, *Shields*, *River Bed*, *Sunflower*, *Rush Hour*, *Tractor Pedestrian Area*, *Blue Sky* and *Station* are reported in Tab. 2, using the VIIDEO [16] no-reference quality metric and frame rates obtained on NVIDIA Titan X. Qualitative inspection of our frames confirm quantitative results, showing pleasant highly detailed frames.

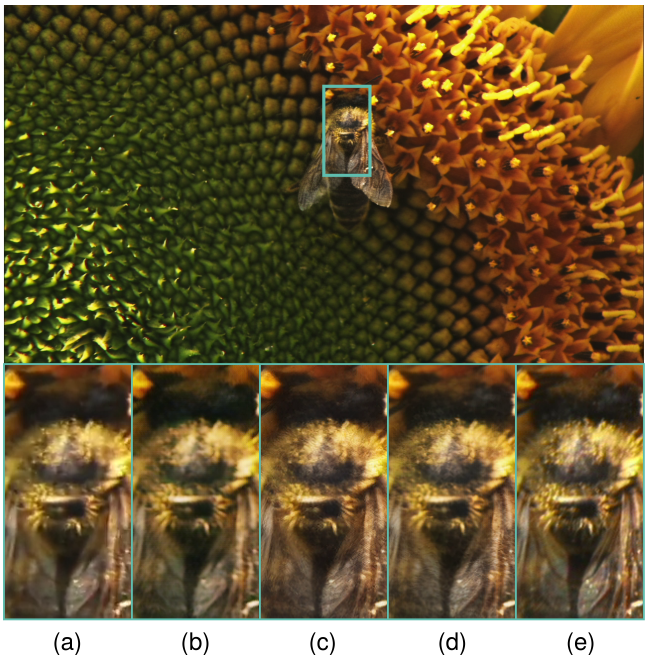

Figure 1: Qualitative comparison of (leftmost) compressed frame with H.264 (CRF 28), (b-d) Very Fast, Fast and Galteri *et al.* [8] networks with (e) uncompressed frame. Large frame obtained by Fast network. Note the fine details of the wings and hairs of the bee obtained by the GAN based approaches, compared to the standard compressed version.

	VIIDEO[16]	FPS@720p
H.264	0.520	-
Very Fast	0.388	42
Fast	**0.350**	20
GAN baseline [8]	0.387	4
Uncompressed	0.276	-

Table 2: No reference quality assessment of our compression artifact removal networks. (lower VIIDEO figure is better, higher FPS is better).

The network has been ported to iOS to perform real-time video enhancement on mobile devices like the iPhone. The light computational cost of the network combined with the recent improvements in deep learning inference hardware i.e. the Apple Neural Engine (APE), was essential to obtain near real-time performance. To enable the use of the APE, the conversion required the removal of padding layers which were integrated into convolutional layers and the choice of fixed input size. For the latter issue, we just converted the network with multiple standard video sizes. The final system works at a frame rate of 12 FPS on an iPhone XS Max.

[1] https://media.xiph.org/video/derf/

REFERENCES

[1] Eirikur Agustsson and Radu Timofte. 2017. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *Proc. of IEEE CVPR Workshops*.

[2] Lukas Cavigelli, Pascal Hager, and Luca Benini. 2017. CAS-CNN: A deep convolutional neural network for image compression artifact suppression. In *Proc. of IJCNN*.

[3] Mengyu Chu, You Xie, Laura Leal-Taixé, and Nils Thuerey. 2018. Temporally Coherent GANs for Video Super-Resolution (TecoGAN). *arXiv preprint arXiv:1811.09393* (2018).

[4] Y. Dar, A. M. Bruckstein, M. Elad, and R. Giryes. 2016. Postprocessing of Compressed Images via Sequential Denoising. *IEEE Transactions on Image Processing* 25, 7 (July 2016), 3044–3058.

[5] Chao Dong, Yubin Deng, Chen Change Loy, and Xiaoou Tang. 2015. Compression artifacts reduction by a deep convolutional network. In *Proc. of ICCV*.

[6] Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. 2007. Pointwise shape-adaptive DCT for high-quality denoising and deblocking of grayscale and color images. *IEEE Transactions on Image Processing* 16, 5 (2007), 1395–1411.

[7] Leonardo Galteri, Lorenzo Seidenari, Marco Bertini, and Alberto Del Bimbo. 2017. Deep Generative Adversarial Compression Artifact Removal. In *Proc. of ICCV*.

[8] L. Galteri, L. Seidenari, M. Bertini, and A. Del Bimbo. 2019. Deep Universal Generative Adversarial Compression Artifact Removal. *IEEE Transactions on Multimedia* (2019), 1–1.

[9] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Proc. of NIPS*.

[10] V. Jakhetiya, W. Lin, S. P. Jaiswal, S. C. Guntuku, and O. C. Au. 2017. Maximum a Posterior and Perceptually Motivated Reconstruction Algorithm: A Generic Framework. *IEEE Transactions on Multimedia* 19, 1 (2017), 93–106.

[11] L. W. Kang, C. C. Hsu, B. Zhuang, C. W. Lin, and C. H. Yeh. 2015. Learning-Based Joint Super-Resolution and Deblocking for a Highly Compressed Image. *IEEE Transactions on Multimedia* 17, 7 (2015), 921–934.

[12] Tero Karras, Samuli Laine, and Timo Aila. 2018. A style-based generator architecture for generative adversarial networks. *arXiv preprint arXiv:1812.04948* (2018).

[13] Tao Li, Xiaohai He, Linbo Qing, Qizhi Teng, and Honggang Chen. 2017. An Iterative Framework of Cascaded Deblocking and Super-Resolution for Compressed Images. *IEEE Transactions on Multimedia* (2017).

[14] Yu Li, Fangfang Guo, Robby T. Tan, and Michael S. Brown. 2014. A Contrast Enhancement Framework with JPEG Artifacts Suppression. In *Proc. of ECCV*.

[15] Xiaojiao Mao, Chunhua Shen, and Yu-Bin Yang. 2016. Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections. In *Proc. of NIPS*.

[16] Anish Mittal, Michele A Saad, and Alan C Bovik. 2016. A completely blind video integrity oracle. *IEEE Transactions on Image Processing* 25, 1 (2016), 289–300.

[17] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. 2018. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In *Proc. of CVPR*.

[18] Pavel Svoboda, Michal Hradis, David Barina, and Pavel Zemcik. 2016. Compression artifacts removal using convolutional neural networks. *arXiv preprint arXiv:1605.00366* (2016).

[19] Zhangyang Wang, Ding Liu, Shiyu Chang, Qing Ling, Yingzhen Yang, and Thomas S Huang. 2016. D3: Deep dual-domain based fast restoration of JPEG-compressed images. In *Proc. of CVPR*.

[20] Tak-Shing Wong, Charles A Bouman, Ilya Pollak, and Zhigang Fan. 2009. A document image model and estimation algorithm for optimized JPEG decompression. *IEEE Transactions on Image Processing* 18, 11 (2009), 2518–2535.

[21] Seungjoon Yang, Surin Kittitornkun, Yu-Hen Hu, Truong Q Nguyen, and Damon L Tull. 2000. Blocking artifact free inverse discrete cosine transform. In *Proc. of ICIP*.

[22] Jaeyoung Yoo, Sang-ho Lee, and Nojun Kwak. 2018. Image Restoration by Estimating Frequency Distribution of Local Patches. In *Proc. of CVPR*.

[23] J. Zhang, R. Xiong, C. Zhao, Y. Zhang, S. Ma, and W. Gao. 2016. CONCOLOR: Constrained Non-Convex Low-Rank Model for Image Deblocking. *IEEE Transactions on Image Processing* 25, 3 (March 2016), 1246–1259.

[24] X. Zhang, R. Xiong, X. Fan, S. Ma, and W. Gao. 2013. Compression Artifact Reduction by Overlapped-Block Transform Coefficient Estimation With Block Similarity. *IEEE Transactions on Image Processing* 22, 12 (2013), 4613–4626.