# Increasing Video Perceptual Quality with GANs and Semantic Coding

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

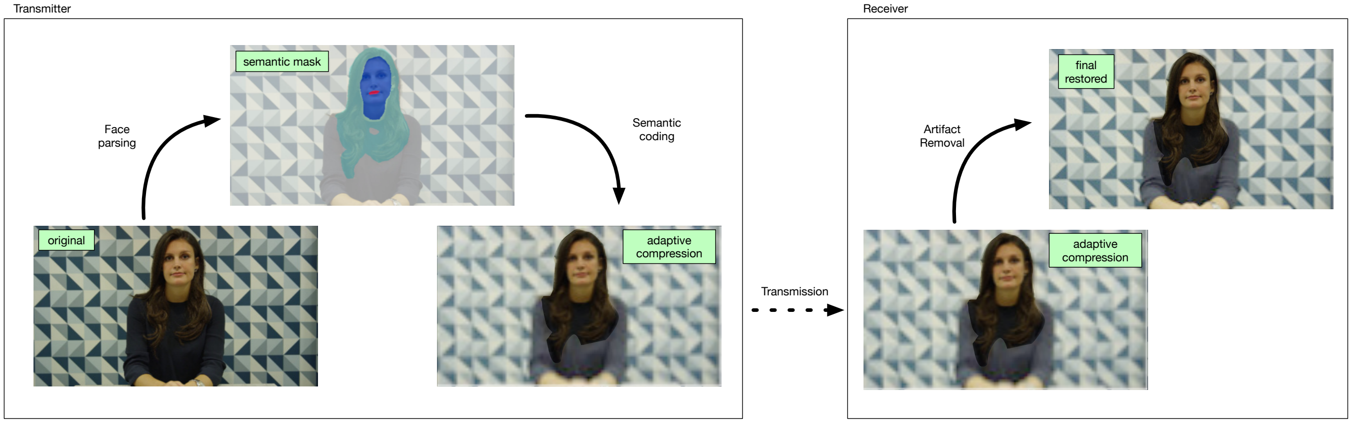

**Figure 1: The system comprised of the transmitter and receiver. The transmitter detects the parts of the image which are important to the perception of a viewer and assign more bandwidth to them. The receiver runs the artifact removal algorithm which recover the missing details accordingly.**

## ABSTRACT

We have seen a rise in video based user communication in the last year, unfortunately fueled by the spread of COVID-19 disease. Efficient low-latency delay of transmission of video is a challenging problem which must also deal with the segmented nature of network infrastructure not always allowing a high throughput. Lossy video compression is a basic requirement to enable such technology widely. While this may compromise the quality of the streamed video there are recent deep learning based solutions to restore quality of a lossy compressed video.

Considering the very nature of video conferencing, bitrate allocation in video streaming could be driven semantically, differentiating quality between the talking subjects and the background. Currently there have not been any work studying the restoration of semantically coded video using deep learning. In this work we show how such videos can be efficiently generated by shifting bitrate with masks derived via computer vision and how a deep generative adversarial network can be trained to restore video quality. Our study shows that the combination of semantic coding and learning based video restoration can provide superior results.

## KEYWORDS

semantic video compression, GANs, video quality enhancement

**ACM Reference Format:**
Leonardo Galteri, Marco Bertini, Lorenzo Seidenari, Tiberio Uricchio, Alberto Del Bimbo. 2020. Increasing Video Perceptual Quality with GANs and Semantic Coding. In *Proceedings of the 28th ACM International Conference on Multimedia (MM '20), October 12–16, 2020, Seattle, WA, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3394171.3413508

## 1 INTRODUCTION

The era of day-to-day videoconferencing has dawned. Stimulated in the recent years by developments in networking like the 5G and modern video codecs, it has seen a dramatic increase with the global spread of COVID-19. People constrained at home by the emergency, talk shows and meetings have all adopted the use of videoconferencing as main media of communication. As a result, global networks have been profoundly impacted with an excessive traffic that they were not prepared to receive. To transmit or store a raw video, it must be compressed to reduce bandwidth and storage requirements. This happens at the cost of the perceived quality which strongly depends on the amount of available bandwidth and the compression algorithm. While video coding algorithms are designed to reduce perceptual quality loss using a model of the human visual system, they do not know video semantics or

*MM '20, October 12–16, 2020, Seattle, WA, USA*
© 2020 Association for Computing Machinery.
ACM ISBN 978-1-4503-7988-5/20/10…$15.00
https://doi.org/10.1145/3394171.3413508

cues on which information is more important to a human viewer. When dealing with specialized tasks, such as video conferencing, the usual bitrate allocation of modern video codecs may not favor the right portion of the frame (e.g. face and upper body of the speaker). The background has so little relevance in this context that some commercial solutions provide features to blur [1] or completely replace the background with a virtual one [2]. For this reason, by combining state-of-the art computer vision techniques with saliency based bitrate allocation, it is possible to drive codecs bitrate to favor content based on semantics and not just on low-level features such as frequency content of the signal. User studies on videos crafted as such have shown little or no effect on the user experience [45].

An obvious solution to reducing bandwidth for transmission is to dramatically cut the quality of compressed video thus reducing bitrate. On the one hand this will "allow" the video call to run smoothly without any delay, on the other hand, the reduced quality in the perceived video will make the user experience, in certain cases, almost unbearable. Recently, solutions to improve image and video quality have been proposed, also running in real-time on tablets and smartphones [14–16]. With these algorithms in play it is possible to increase quality and resolution of inbound highly compressed and subsampled videos.

In this work we provide the following contributions:

- We design a system for streaming talking humans efficiently, combining semantic coding of a source video with a deep learning based image restoration process.
- We provide an extensive evaluation using both full-reference and no-reference image assessment metrics, showing that our GAN trained on semantically coded video is able to improve the overall quality better than a generic image enhancement network.

Currently, to the best of our knowledge, learning based image enhancement methods have not been applied onto videos which have been semantically encoded. Moreover we show that our system is able to provide comparable quality for videos talking humans with a third of the bandwidth.

## 2 PREVIOUS WORK

Semantic video coding is regarded as a straightforward solution for bandwidth requirements. The basic idea is to identify objects which are more perceptually relevant for the viewer and improve their appearance increasing bit allocation adaptively. We can frame two main lines of research: visual saliency [4, 26] and object [12, 45] based video coding. In the former approach some function of the image is computed pixelwise irrespectively of the semantic content of the image. Such function measure the relevance of frame regions and is used to modulate bitrate. Object based video instead assumes some form of segmentation has been applied to obtain masks of relevant objects. This approach requires using robust object segmentation which are nowadays deployable at high efficiency [8, 18].

*Video and image restoration.* Recently, learning based image enhancement has been proposed [9, 10, 13, 14, 21, 28, 29, 39, 44, 46]. Such approaches, learn deep convolutional architectures to transform images corrupted by artifacts into high quality ones.

The first work employing CNNs for compression artifact removal is [10]. Their network design is specialized for JPEG compression, while more recent works [9, 39] employ general purpose architectures all sharing some common features such as residual learning and skip connections bringing the benefit of allowing several layers of representation and propagating information from earlier layers to the final reconstruction directly. Interestingly, most perceptually satisfying results are obtained using Generative Adversarial Networks [14]. In [13, 14] Galteri et al. show that GAN based image restoration can be performed on various encoders even in an agnostic setting by predicting coding parameters. All of the above algorithms have never been used in conjunction with semantic video coding. In this work we present GAN models to increase quality of semantically coded faces.

*Video and image compression.* Semantic video coding approaches can be used in very different domains, such as airplane cockpits [30], sport videos [6], drone videos [41], vehicles [3], and surveillance videos [5]. Some preliminary effort has been made to perform video and image coding using neural networks [36, 37]. These approaches are currently not deployable with satisfying visual results due to an unbearable computational footprint. Moreover, fully learned compression, requires the standardisation and diffusion of a novel technology thus raising a high market barrier to entry. This can be mitigated if the decoding end of the pipeline is kept to standard. As partially reviewed in [35] there are two main strategies to improve video quality while still relying on standardized encoding solutions: pre-processing based and post-processing based. As an example, Talebi *et al.* [40] proposed an hybrid approach to improve the quality of compressed images. Instead of relying on a deep network for encoding and decoding they learn a deep network for pre-processing images before standard JPEG compression. Training objective minimize entropy and image distortion jointly for a given JPEG quality factor. In this work we are the first to apply state-of-the-art GAN based image restoration to video that have been compressed with a semantic cue, thus intervening on both ends of the coding pipeline, which is still based on standard video codecs.

*Quality metrics.* It is important to also consider how images appear at the end of the encoding-decoding pipeline. In our scenario a reference image is available allowing to also perform Full-reference image quality assessment. The recent work from Blau and Michaeli [7] has shown that there is a rate-distortion-perception trade-off showing that optimizing the statistical similarity of source and decoded images will increase the signal distortion rate. This is in line with the copious amount of results that shows how images ranked higher by humans obtains a lower score according to SSIM and PSNR metrics. For these reasons a lot of work has been dedicated to obtain more reliable metrics for image quality assessment [23, 32, 33, 48]. In our work we will rely on modern LPIPS metric as a full-reference evaluation and BRISQUE for a no-reference image quality assessment.

## 3 THE PROPOSED METHOD

Our approach is based on the idea that a compression artefact removal method can restore videos with a better perceived visual quality by exploiting semantically encoded video. We first describe how the semantic video encoding is performed on the transmitter.

Then, we report the GAN-based video restoration approach to improve the perceptual visual quality on the receiver party.

## 3.1 Semantic video encoding

The main idea of semantic video encoding is to allocate more bits to the regions that depict semantic content of interest for the viewer, to the detriment of background. Ideally, the amount of bits should be enough to maximize the perceptual quality of the objects of interest for a viewer. Semantic video encoding is related to saliency based video encoding [26, 27] as they both consider regions that should be stored with a higher amount of data. Nonetheless they aim at different targets. Semantic encoding aims at transferring the high level semantic content that is of most interest of the viewer, regardless of any other element of background. The saliency based encoding, instead, has no specific knowledge of objects of the scene. It aims at transferring the content which is most probably observed by the eyes of a viewer, regardless of its importance.

To perform the encoding, we construct a semantic mask for each frame where we label each pixel as foreground (i.e. pixels of regions that are allotted more bits) or background. Depending on the domain, the foreground may be different. In our considered context of a video conference application, the foreground is the speaking person, more specifically its face. Hence, we employ the popular BiSeNet [47] image segmentation method, trained on CelebAMask-HQ [25] to perform face parsing. We label each pixel detected as a part of face and neck as foreground, the remainder as background.

The final video is encoded using a h.264 encoder which has been modified to allot a predetermined $P$ percentage of a given bitrate to an input mask. We employ the implementation of [26] which uses a non-trivial estimation of macroblock sizes with respect to the quantization of parameter of h.264 constant quantizer.

## 3.2 Video restoration

Most restoration approaches based on deep learning tackle the artifact removal problem trying to minimize the squared pixel-wise Euclidean distance between a reference raw frame $I$ and the generated output $I^R$ from a compressed input $I^C$. However, this kind of training strategy leads to feeble restored images as they appear often blurry and lacking important details. Besides, the h.264 encoder typically contains a strong loop de-blocking filter at the end of the compression pipeline, which leads to producing blurry frames, so that using an MSE based neural network to restore the images is even less effective.

Generative Adversarial Networks have been broadly used for both restoration and enhancement tasks to solve the aforementioned issues. The GAN framework tries to estimate a model distribution that approximate a target distribution, and it comprises two distinct entities, a generator and a discriminator. In this setup, the aim of the generator is to produce the model distribution given some noisy input and the role of the discriminator is to discern the model distribution from the target one. The two networks are trained one after another while gradually the distance between the model distribution and the generator decreases.

Since we do not want to generate completely novel images from the model distribution, but we rather want to restore some distorted data, we need to condition the training procedure of the GAN accordingly. Therefore, we feed the discriminator with real samples $I|I^C$ and fake samples $I^R|I^C$ where the operator $\cdot|\cdot$ defines the channel-wise concatenation of the inputs.

*Architectures.* The architecture of our generator is based on [13], which is composed mostly of residual blocks and convolutional layers, with no Batch-Normalization. Differently from [13] we train the network to learn the residual image, hence there is a skip-connection between the input image and the restored output. Using this scheme we reduce the overall training time and improve its stability. We choose both input and output values to be in the $[0, 1]$ range. We employ the most common architecture for our discriminator, a sequence of convolutional layers followed by LeakyReLU activation, with a final output dense layer.

*Losses.* Following the results of [42] we choose the Relativistic GAN [20] instead of the standard GAN setup to get better reconstruction outputs. Here, the key idea is to drive the discriminator to estimate the probability that a ground truth image $I$ is relatively more realistic than a generated one $I^R$. We define $D(I, I^R) = \sigma(C(I) - \mathbb{E}_{I^R}\left[C(I^R)\right])$ as the output of the relativistic discriminator, where $\sigma$, $C(.)$ and $\mathbb{E}_{I^R}\,[.]$ stand for the sigmoid activation, the dense layer output of the discriminator and the average for all reconstructed images in the mini-batch, respectively. The discriminator loss is defined as:

$$\begin{aligned} L_D = &-\mathbb{E}_I[log(D(I, I^R))] \\ &-\mathbb{E}_{I^R}[1 - log(D(I^R, I))] \end{aligned} \tag{1}$$

and the adversarial loss for the generator as:

$$\begin{aligned} L_{Adv} = &-\mathbb{E}_I[1 - log(D(I, I^R))] \\ &-\mathbb{E}_{I^R}[log(D(I^R, I))] \end{aligned} \tag{2}$$

Following the contribution of many perceptual-driven approaches [11, 17, 19, 24] to improve the visual quality of restored outputs, we use a loss based on perceptual similarity in our adversarial training. We minimize the distance between images by projecting $I$ and $I^R$ on a feature space with a differentiable function $\phi$ and taking the L1 distance between the two different representations:

$$L_{perceptual} = \mathbb{E}_{(I,I^R)}\left[||\phi(I) - \phi(I^R)||\right] \tag{3}$$

In this work we implement the VGG-19 network to extract the feature representations, adopting the output taken from the fourth convolutional layer of the fifth block before the ReLU activation. For convenience we name the standard perceptual loss based on the VGG-19 network as $L_{VGG}$.

We define a more effective perceptual loss by realizing that our data to be reconstructed is not homogeneous. As a matter of fact, the semantic encoding partitions the image in two regions that differs from content (background/faces) and quality (low-quality/high-quality). Therefore, the network needs to learn the reconstruction of different parts according to separate objectives using the semantic masks computed by the face parser. We define $M$ as the foreground binary mask and $\overline{M}$ as the background mask that is computed by the logical negation of $M$.

We keep the VGG loss for the background, but we limit its computation to the parts of the image where $\overline{M}$ values are not zero.

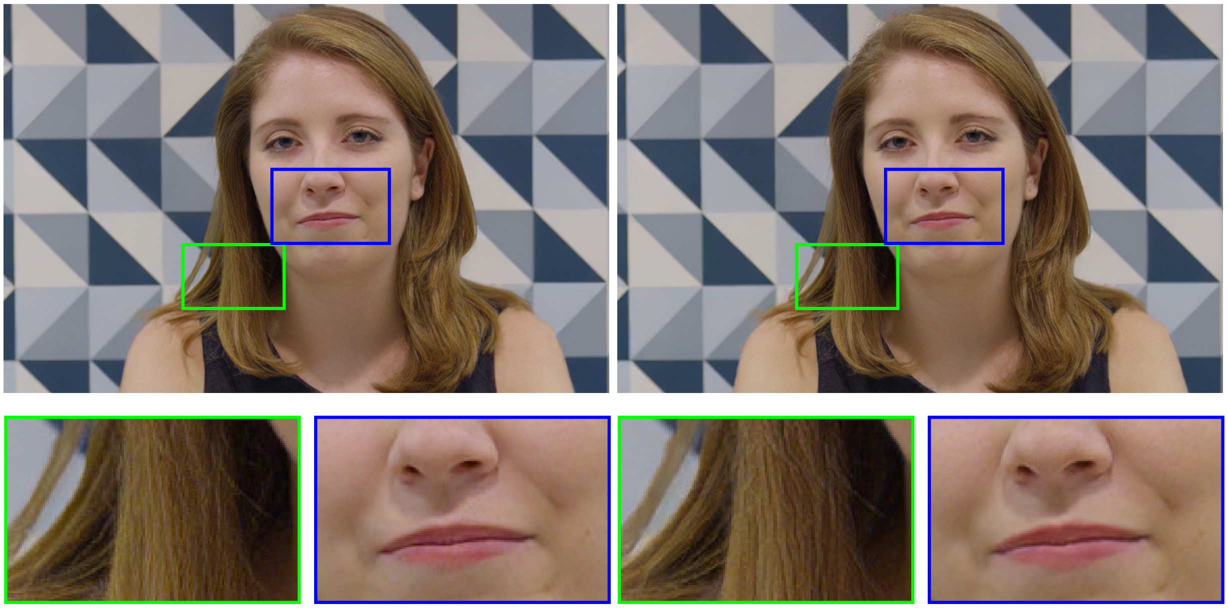

**Figure 2: Examples of frames restored using the two losses: *Left)* GAN with VGG perceptual loss; *Right)* GAN with combined VGG background and VGG-Face foreground loss. Note how the mouth (especially lower lip) and the nasolabial folds are more detailed and with less artifacts; the zoomed area of the hairs of the right image has more details and natural texture. See supplementary materials for higher quality image.**

We name the perceptual loss based on the VGG-19 network for the background as $L_B$:

$$L_B = \mathbb{E}_{(I,I^R)} \left[ ||VGG(I \odot \overline{M}) - VGG(I^R \odot \overline{M})|| \right] \tag{4}$$

where ⊙ stands for element by element multiplication. Since the foreground is composed of human faces, we choose to adopt a different extractor to handle this specific category of features. The logical choice is to extract such features from a pre-trained network that has processed millions of face images, that is VGG-Face [34]. As VGG-Face is based on VGG-16 backbone, we extract the output taken from the third convolutional layer of the fifth block before the ReLU activation for the loss computation. Under these assumptions, we define the perceptual loss constrained to the foreground as:

$$L_F = \mathbb{E}_{(I,I^R)} \left[ ||VGGFace(I \odot M) - VGGFace(I^R \odot M)|| \right] \tag{5}$$

The total loss for the generator is:

$$L_G = L_B + L_F + \lambda L_{Adv} \tag{6}$$

where $\lambda$ is a fixed coefficient to balance the contribution of the adversarial loss.

*Training Details.* In all our configurations we extract 8 random 256 × 256 patches from the training data with random left-right flipping. During the training phase we use Adam [22] as optimizer for both generator and discriminator with momentum 0.9 and a learning rate of $10^{-4}$ for the first 10 epochs. We halve the learning rate every other 10 epochs for an overall amount of 40. We have trained our reconstruction models with PyTorch and a NVIDIA Titan Xp GPU.

## 4 EXPERIMENTAL RESULTS

### 4.1 Dataset

We have used Deep Fake Detection dataset [38], that is composed of 363 high resolution and high quality videos depicting different activities performed by 28 actors; we have used the raw (compression rate factor 0) versions of the original sequences (∼ 200GB size). We have then selected 55 videos of actions in which the actor is talking while facing the camera as in a setup of a video conference (i.e. "podium speech" and "talking against wall" scenes) for an overall size of ∼ 40 GB and a duration of ∼ 40 minutes. The first 22 identities have been used for training and the last 6 for testing.

### 4.2 Video quality metrics

Since we are dealing with image compression and restoration tasks, a reference image is available to perform evaluation. Full-reference image quality assessment uses a reference version of an image to compute a similarity. The popular SSIM (Structural SIMilarity) [43] is a metric of structural similarity that is more consistent than MSE and PSNR with perceived quality. The SSIM index varies between -1 and 1, where 1 indicates perfect structural similarity, while 0 indicates no structural similarity. However, it must be noted that, as reported in [23], many existing image quality algorithms like SSIM are unreliable on GAN generated content, since images generated by GANs may appear quite realistic and similar to an original, yet may match it poorly based on simple pixel comparisons; metrics based on "naturalness" are more suitable in this case. Nevertheless, we report results using this metric due to its widespread use.

Differently from SSIM, BRISQUE (Blind/Referenceless Image Spatial Quality Evaluator) [31] is a no-reference metric, thus it does not require a reference image to evaluate the quality of the compressed version. BRISQUE evaluates natural scene statistics to quantify losses of “naturalness" due to distortions like those introduced by compression. A smaller BRISQUE score indicates better perceptual quality. We report results using this metric as a way to measure the naturalness of an image, that may be associated in our use case to how natural looks a face and its features.

Finally, we have used the recent LPIPS (Learned Perceptual Image Patch Similarity) [48] metric, a novel full-reference metric that evaluates the distance between image patches, based on deep features; the authors have shown that LPIPS outperforms traditional metrics like SSIM by a large margin in a two alternative forced choice (2AFC) test, that asks which of two distorted images is more similar to a reference. Higher LPIPS score means that two patches are more different perceptually, a lower score means they are more similar. We report results using this metric to evaluate the quality of reconstruction w.r.t. the high quality version of videos using a metric that is able to capture better distortions as perceived by the human visual system. Typically LPIPS measures are in contrast with SSIM, i.e. distortions that look more similar for SSIM are considered distant in LPIPS.

The SSIM and LPIPS full-reference metrics have been computed comparing the compressed and reconstructed frames to the frames obtained from the raw (CRF 0) videos; BRISQUE has been computed directly on patches of the compressed and reconstructed frames, since it does not require any comparison. Measures have been computed considering only patches obtained from automatically detected faces and from patches over the whole frame.

## 4.3 Semantic video coding

In the first set of experiments we evaluate the effect of semantic video coding at varying bitrate and with different percentages of bitrate allocation to the semantically relevant parts of the frame, i.e. face parts. The quality of the compressed videos is evaluated on the patches within the bounding box of the detected faces and over the whole frame.

Table 1 and Table 2 report quality metrics for videos compressed with relatively high bitrates of 1000 kb/sec. The value reported in the first column reports the percentage of the bitrate allocated to the semantically salient regions, i.e. the mask generated by the segments of the face; when the value is 0 then no semantic video coding is used and the standard h.264 coding is used. Tab. 1 values have been computed on patches of the face, while values of Tab. 2 have been computed over the whole frame. It can be observed that as the percentage of bitrate allocated to semantically salient regions increases all the metrics improve when considering the quality of faces. Considering the whole frame the best LPIPS results are obtained for a saliency allocation of 15%, and second best for a value of 25%. Instead, the best BRISQUE and SSIM results are obtained without using saliency, as the encoder has enough bitrate available to encode in high quality all the frame and is free to allocate bandwidth wherever necessary; it must be also considered that there are many more patches belonging to the background than to the face and that the background is relatively uniform.

**Table 1: Quality metrics for higher bitrate videos (1000Kbps); metrics computed on face patches only. Best results highlighted in bold, second best are underlined.**

| % sal. BR | LPIPS | BRISQUE | SSIM | Dim. (KB) |
|---:|---:|---:|---:|---:|
| 0 | 0,052 | 32,95 | 93,45 | 301.616 |
| 10 | 0,043 | 29,61 | 94,15 | 269.932 |
| 15 | 0,038 | 27,73 | 94,59 | 268.788 |
| 25 | **0,034** | **25,58** | **94,91** | 267.308 |

**Table 2: Quality metrics for higher bitrate videos (1000Kbps); metrics computed on whole frame. Best results highlighted in bold, second best are underlined.**

| % sal. BR | LPIPS | BRISQUE | SSIM | Dim. (KB) |
|---:|---:|---:|---:|---:|
| 0 | 0,417 | **60,18** | **97,37** | 301.616 |
| 10 | 0,413 | 62,14 | 97,22 | 269.932 |
| 15 | **0,408** | 63,11 | 97,05 | 268.788 |
| 25 | 0,411 | 63,73 | 96,71 | 267.308 |

Table 3 and Table 4 report quality metrics for a more challenging setup, where videos are compressed with relatively low bitrates of 400 kb/sec, resulting in an overall dimension of a third of the previous one. As in the previous tables, we include a version that does not use semantic coding, i.e. the percentage of bitrate allocated to semantically salient regions is 0. Tab. 3 values have been computed on patches of the face, while values of Tab. 4 have been computed over the whole frame. As it can be expected the values are worst than those obtained for higher bitrates reported in the two previous tables. Also in this case, using saliency improves image quality computed on the face patches, but it must be noted that in this more challenging scenario also the BRISQUE metric is better with saliency when evaluating over the whole frame; only the older SSIM metric is better without saliency, but only by a very small value. Overall coding with a 15-25% bitrate assigned to salient regions provides the best results when coding at lower bitrates.

**Table 3: Quality metrics for lower bitrate videos (400Kbps); metrics computed on face patches only. Best results highlighted in bold, second best are underlined.**

| % sal. BR | LPIPS | BRISQUE | SSIM | Dim. (KB) |
|---:|---:|---:|---:|---:|
| 0 | 0,078 | 38,02 | 91,32 | 117.540 |
| 10 | 0,068 | 35,44 | 92,24 | 118.288 |
| 15 | 0,063 | 33,92 | 92,62 | 118.596 |
| 25 | **0,061** | 32,34 | **92,73** | 118.968 |
| 35 | 0,062 | 31,45 | 92,58 | 119.204 |
| 45 | 0,063 | **30,67** | 92,46 | 119.308 |

## 4.4 Improving video quality

In this set of experiments we evaluate the quality of the proposed quality improvement method described in Sect. 3.2, applied both to videos compressed semantically and without semantic compression. Similarly to the first set of experiments, visual quality is computed on the patches within the bounding box of the detected faces and over the whole frame. In these experiments the dimension of files is not reported since the proposed approach performs quality improvement when decoding frames, so the size of the files is the

**Table 4: Quality metrics for lower bitrate videos (400Kbps); metrics computed on whole frame. Best results highlighted in bold, second best are underlined.**

| % sal. BR | LPIPS | BRISQUE | SSIM | Dim. (KB) |
|---|---|---|---|---|
| 0 | 0,411 | 64,79 | **96,60** | 117.540 |
| 10 | 0,412 | 65,01 | 96,40 | 118.288 |
| 15 | 0,410 | 64,74 | 96,17 | 118.596 |
| 25 | **0,409** | **64,24** | 95,66 | 118.968 |
| 35 | 0,416 | 64,50 | 95,11 | 119.204 |
| 45 | 0,421 | 64,97 | 94,56 | 119.308 |

**Table 5: Quality metrics for improved versions of lower bitrate videos (400Kbps); metrics computed on face patches only. Best results highlighted in bold, second best are underlined. Changes w.r.t. compressed versions (Tab. 3) reported in parentheses, +/- stands for improvement/deterioration.**

| % sal. BR | LPIPS | BRISQUE | SSIM |
|---|---|---|---|
| 0 | 0,047 (+40,00%) | 15,44 (+59,38%) | 89,04 (-2,49%) |
| 10 | 0,042 (+38,24%) | 13,55 (+61,77%) | 89,85 (-2,59%) |
| 15 | **0,040** (+36,25%) | 12,90 (+61,95%) | 90,18 (-2,64%) |
| 25 | 0,041 (+32,79%) | 12,12 (+62,51%) | **90,23** (-2,70%) |
| 35 | 0,044 (+29,03%) | 11,69 (+62,82%) | 90,04 (-2,74%) |
| 45 | 0,046 (+27,45%) | **11,25** (+63,31%) | 89,82 (-2,86%) |

**Table 6: Quality metrics for improved versions of lower bitrate videos (400Kbps); metrics computed on whole frame. Best results highlighted in bold, second best are underlined. Changes w.r.t. compressed versions (Tab. 4) reported in parentheses, +/- stands for improvement/deterioration.**

| % sal. BR | LPIPS | BRISQUE | SSIM |
|---|---|---|---|
| 0 | 0,406 (+1,1%) | **24,01** (+62,9%) | **95,19** (-1,46%) |
| 10 | 0,405 (+1,8%) | 24,39 (+62,5%) | 95,07 (-1,38%) |
| 15 | 0,408 (+0,4%) | 24,79 (+61,7%) | 94,93 (-1,29%) |
| 25 | 0,406 (+0,7%) | 24,93 (+61,2%) | 94,57 (-1,14%) |
| 35 | **0,403** (+3,2%) | 25,08 (+61,1%) | 94,17 (-0,99%) |
| 45 | 0,410 (+2,6%) | 24,53 (62,2%) | 93,69 (-0,92%) |

**Table 7: Quality metrics for improved versions of lower bitrate videos (400Kbps), using the loss that combines VGG background and VGG-Face foreground losses; metrics computed on face patches only. Best results highlighted in bold, second best are underlined. Changes w.r.t. compressed versions (Tab. 3) reported in parentheses, +/- stands for improvement/deterioration.**

| % sal. BR | LPIPS | BRISQUE | SSIM |
|---|---|---|---|
| 0 | 0,046 (+40,69%) | 18,31 (+51,83%) | 89,23 (-2,28%) |
| 10 | 0,041 (+39,37%) | 16,24 (+54,16%) | 90,07 (-2,35%) |
| 15 | **0,039** (+37,81%) | 15,39 (+54,63%) | 90,41 (-2,38%) |
| 25 | 0,040 (+34,20%) | 14,45 (+55,33%) | **90,47** (-2,43%) |
| 35 | 0,042 (+31,52%) | 13,86 (+55,93%) | 90,29 (-2,47%) |
| 45 | 0,045 (+29,37%) | **13,26** (+56,76%) | 90,07 (-2,59%) |

same of those reported in the previous section. Table 5 reports quality metrics computed over face patches, while Table 6 results have been obtained computing them over the whole frame. Quality improvement has been applied to the low bitrate versions of videos (400 kb/s), since they are the more challenging and this setup is more relevant for the video chat domain. The tables report also the variation w.r.t. the corresponding metrics of Tab. 3 and Tab. 4.

Comparing the values of Tab. 5 with those of Tab. 3, shows that the proposed approach greatly improves image quality of the faces; the GAN is able to add realistic face details and the LPIPS values are now in the same range of those of videos encoded at 1000 kb/s, showing that using our proposed approach is possible to encode at less than half the bitrate while keeping the same quality. It is interesting to note that the proposed approach reduces the difference in LPIPS score between semantic coding and standard coding: this means that the proposed restoration is effective even if no semantic coding is used. Similarly to Tab. 3 using a 15-25% allocation for semantic saliency results in the best performance in terms of LPIPS full-reference metrics. Considering Tab. 6, the best values of the more reliable LPIPS metric are obtained on videos compressed using saliency. Comparing Tab. 6 with Tab. 4, shows that also in this case metrics are improved, although with a lesser extent than when considering faces only. It is interesting to note that BRISQUE metric improves greatly, with a ratio similar to that of faces only. We explain this due to the fact that the GAN-based approach adds "natural" details to the background in the image.

As expected, and noted in other works that applied generative approaches to image quality improvement like [13, 23], the SSIM metric shows a small decrease in both cases. This is due to the fact that GANs "hallucinate" details, thus signal-based full-reference metrics are unable to account for the improvements. As an example consider a GAN that restores "naturally looking" hairs in positions that are slightly off-set w.r.t. their actual position in the raw videos: the SSIM metric would result in a lower value, while a metric like LPIPS correctly results in an improvement.

Table 7 reports results obtained using the GAN approach that combines VGG background and VGG-Face foreground loss. Quality metrics computed over face patches, and comparing them with those of Tab. 5 we can observe that reference metrics are improved, with the reliable LPIPS and the older SSIM. Instead BRISQUE, although greatly improving with respect to the compressed video has a smaller reduction with respect to the other GAN approach. This can be explained by the fact that this GAN is able to better recover details that are more similar to the uncompressed frames, thus the reference-based score is better, while adding less high frequency details that make the image appear more "natural" according to BRISQUE algorithm. Overall, results show that using an allocation of 15-25% of the bitrate to the salient regions results in the best performance.

*4.4.1 Qualitative examples.* All the figures and Fig. 2 are reported in higher quality in the supplementary materials. Readers are suggested to refer to them in order to better appreciate the differences.

All the figures are frames from videos compressed at 400 kb/s and compare different versions of the same frame. In Figure 3 the left image shows a frame compressed using standard h.264; many details have compression artifacts, such as the eyes and the hairs, that have lost their finer structure. The bottom lip shows some ringing artifacts in the mouth, there are blockiness artifacts on the skin of face and neck. Also the background shows false colors and bands, e.g. in the upper left part. The middle image shows a frame compressed using h.264 and saliency, assigning 15% of bitrate to the parts of the image containing face, hair and neck. In this case the eyes are more detailed, especially the left one, the skin of the face has less blocking artifacts and hairs are more detailed. The right image shows a frame reconstructed from the middle image using the proposed GAN approach. It can be noticed that mouth is more detailed, the skin of face and neck is smoother and with even less blockiness, hairs are more detailed, especially the tip near the arms; also many artifacts in the background have been eliminated (e.g. in the upper left part). In Figure 4 the left image shows a frame compressed using standard h.264; similarly to Fig. 3 finer details like eyes, eyebrows, facial hair and hair have lost details. The skin has blockiness artifacts both in the face and arms, and the shirt has the same issues. The middle image shows a frame compressed using h.264 and saliency, assigning 15% of bitrate to the parts of the image containing face, hair and neck. This results in a smoother skin, more details for eyes and hair. The skin of the arms is still blocky, and the background has about or even more artifacts than the top image. The right image shows a frame reconstructed from the middle image using the proposed GAN approach. Hair are more detailed, and also facial hair and eyebrows. The arms have a smoother skin, the shirt has more details and artifacts in the background have been reduced. In Figure 5 the left image shows a frame compressed using standard h.264; many features of the face have been distorted by compression artifacts, such as the eyes mouth and teeth. Hairs show blockiness artifacts, wall and dress have posterization effects. The middle image shows a frame compressed using h.264 and saliency, assigning 15% of bitrate to the parts of the image containing face, hair and neck. In this case the eyes and mouth are more detailed, the skin of the face has less blocking artifacts and hairs are more detailed. The right image shows a frame reconstructed from the middle image using the proposed GAN approach. It can be noticed that cheeks are smoother, hairs have more details, the posterization of dress and wall has been eliminated.

## 5 CONCLUSIONS

In this work we have evaluated the improvement in perceptual video quality that can be obtained by combining two approaches: semantic video coding by the transmitter and GAN-based compression artefact removal by the receiver. The method has been applied to videos that simulate the use case of video chats, coding semantically salient parts like face and neck with a predefined percentage of the total bitrate. Experimental results show that each approach, when applied alone improves objective quality metrics like LPIPS and BRISQUE that evaluate perceptual quality and naturalness of images. The experiments show also that the combination of both approaches results in increased improvements, and the it is possible to obtain a perceptual quality similar to that obtained using three

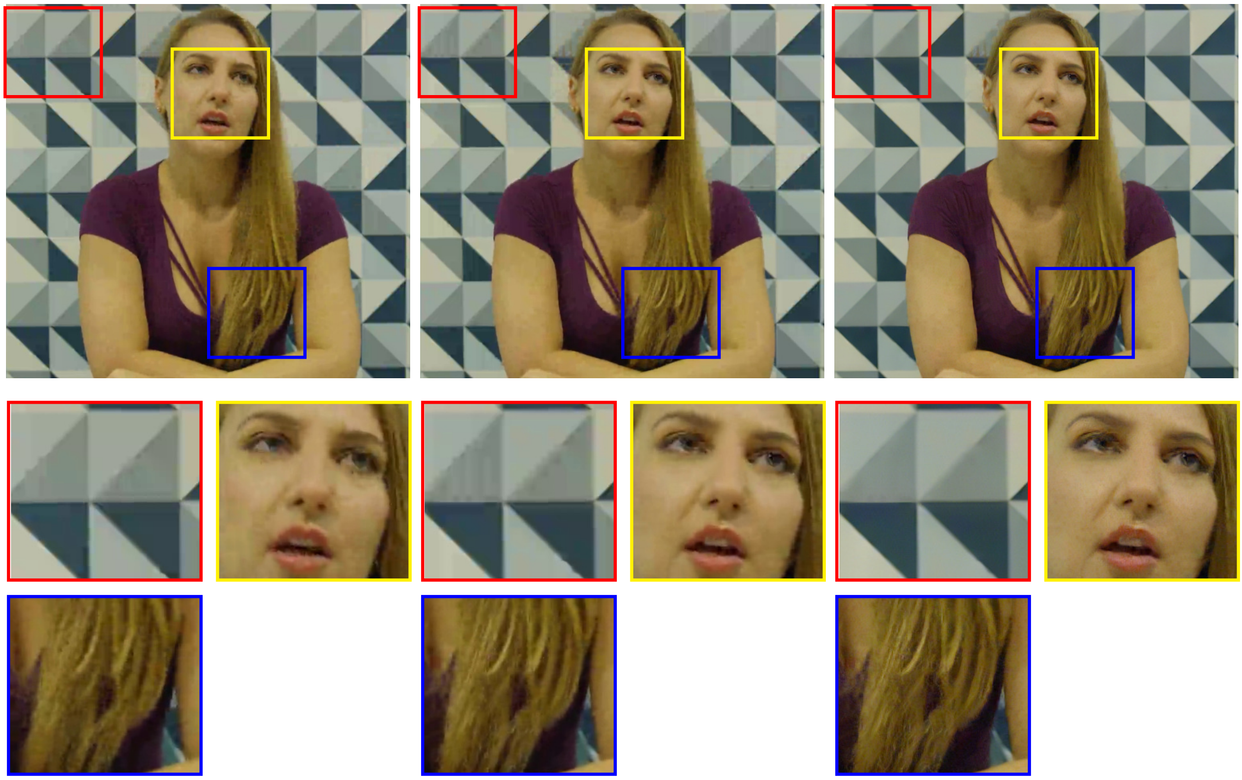

**Figure 3: Examples of frames compressed with: *Left)* standard h.264; *Mid)* h.264 and saliency (15%); *Right)* saliency and proposed GAN improvement. Details like, eyes, mouth, skin, hair and even tiles in the background are progressively improved.**

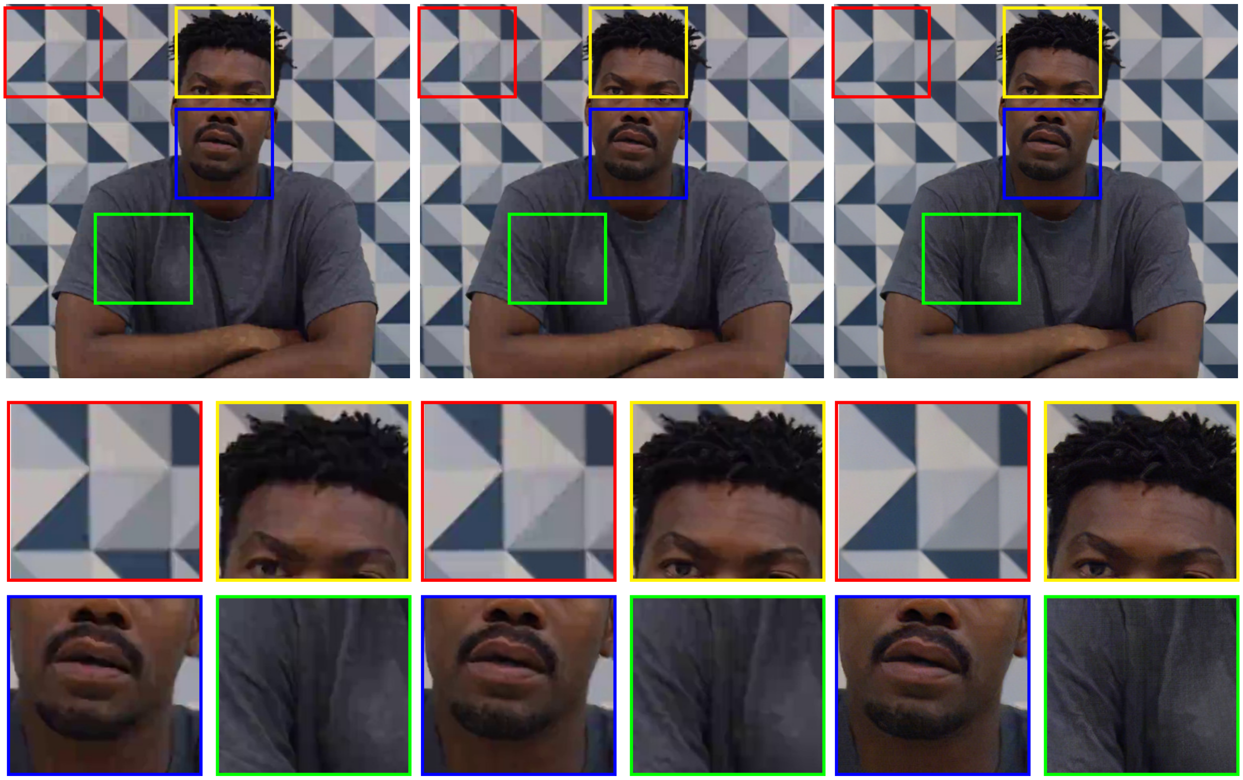

**Figure 4: Examples of frames compressed with: *Left)* standard h.264; *Mid)* h.264 and saliency (15%); *Right)* saliency and proposed GAN improvement. Note how details like, eyes, mouth, skin, hair, shirt and background tiles are progressively improved.**

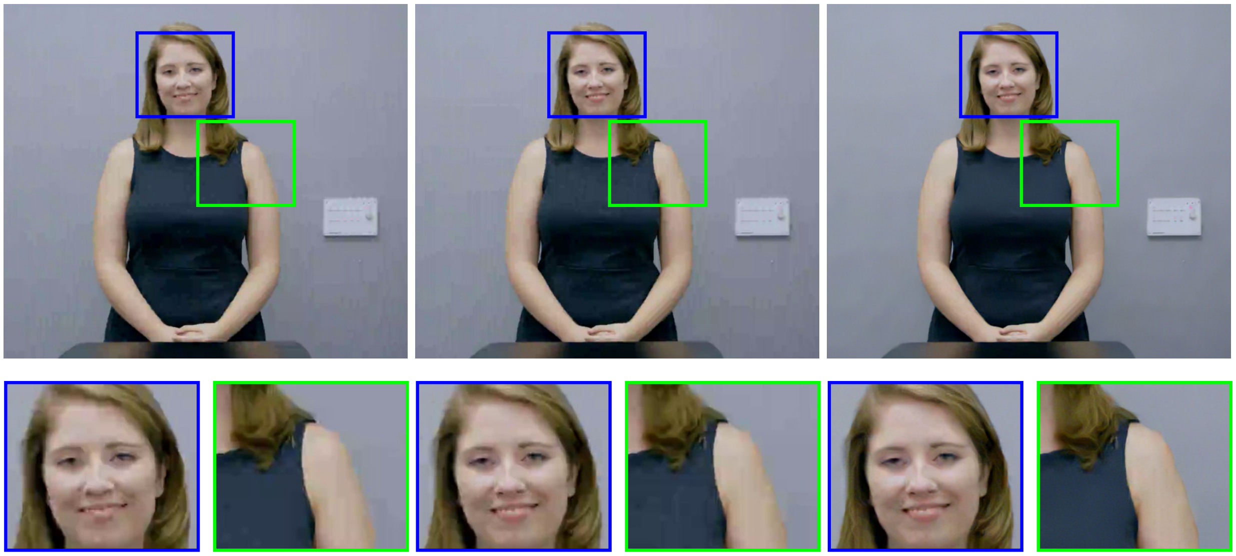

**Figure 5: Examples of frames compressed with: *Left)* standard h.264; *Mid)* h.264 and saliency (15%); *Right)* saliency and proposed GAN improvement. Note how details like, eyes, mouth, skin, hair, dress and wall are progressively improved.**

times the bandwidth. Using a GAN-based approach allows not only to eliminate compression artefacts but also to recreate plausible natural details like hair and facial features.

## ACKNOWLEDGMENTS

We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan Xp GPU used for this research.

## REFERENCES

[1] [n.d.]. Skype video conferencing application. http://www.skype.com.

[2] [n.d.]. Zoom video conferencing application. http://www.zoom.us.

[3] Noor Al-Shakarji, Filiz Bunyak, Hadi Aliakbarpour, Guna Seetharaman, and Kannappan Palaniappan. 2019. Multi-Cue Vehicle Detection for Semantic Video Compression in Georegistered Aerial Videos. In *Proc. of (CVPR) Workshops.*

[4] A Diana Andrushia and R Thangarjan. 2018. Saliency-based image compression using walsh–hadamard transform (WHT). In *Biologically rationalized computing techniques for image processing applications*. Springer, 21–42.

[5] Andrew D. Bagdanov, Marco Bertini, Alberto Del Bimbo, and Lorenzo Seidenari. 2011. Adaptive Video Compression for Video Surveillance Applications. In *Proc. of International Symposium on Multimedia.* https://doi.org/10.1109/ISM.2011.38

[6] Marco Bertini, Alberto Del Bimbo, Andrea Prati, and Rita Cucchiara. 2006. Semantic Adaptation of Sport Videos with User-centred Performance Analysis. *IEEE Transactions on Multimedia* 8, 3 (Jun 2006), 433–443. https://doi.org/10.1109/TMM.2006.870762

[7] Yochai Blau and Tomer Michaeli. 2019. Rethinking lossy compression: The rate-distortion-perception tradeoff, In Proc. of ICML. *arXiv preprint arXiv:1901.07821.*

[8] Daniel Bolya, Chong Zhou, Fanyi Xiao, and Yong Jae Lee. 2019. YOLACT: real-time instance segmentation. In *Proc. of International Conference on Computer Vision.* 9157–9166.

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

[10] Chao Dong, Yubin Deng, Chen Change Loy, and Xiaoou Tang. 2015. Compression artifacts reduction by a deep convolutional network. In *Proc. of International Conference on Computer Vision.*

[11] A. Dosovitskiy and T. Brox. 2016. Generating Images with Perceptual Similarity Metrics based on Deep Networks. In *Proc. of NIPS.*

[12] Leonardo Galteri, Marco Bertini, Lorenzo Seidenari, and Alberto Del Bimbo. 2018. Video compression for object detection algorithms. In *Proc. of International Conference on Pattern Recognition.* IEEE, 3007–3012.

[13] Leonardo Galteri, Lorenzo Seidenari, Marco Bertini, and Alberto Del Bimbo. 2017. Deep Generative Adversarial Compression Artifact Removal. In *Proc. of International Conference on Computer Vision.*

[14] 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.

[15] Leonardo Galteri, Lorenzo Seidenari, Marco Bertini, and Alberto Del Bimbo. 2019. Towards Real-Time Image Enhancement GANs. In *Proc. of International Conference on Analysis of Images and Patterns (CAIP).* IAPR.

[16] Leonardo Galteri, Lorenzo Seidenari, Marco Bertini, Tiberio Uricchio, and Alberto Del Bimbo. 2019. Fast Video Quality Enhancement Using GANs. In *Proc. of ACM Multimedia* (Nice, France) *(MM '19).* Association for Computing Machinery, New York, NY, USA, 1065–1067. https://doi.org/10.1145/3343031.3350592

[17] Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge. 2015. Texture synthesis and the controlled generation of natural stimuli using convolutional neural networks. *CoRR* abs/1505.07376 (2015).

[18] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. 2017. Mask r-cnn. In *Proc. of International Conference on Computer Vision.* 2961–2969.

[19] Justin Johnson, Alexandre Alahi, and Fei-Fei Li. 2016. Perceptual Losses for Real-Time Style Transfer and Super-Resolution. *CoRR* abs/1603.08155 (2016). http://arxiv.org/abs/1603.08155

[20] Alexia Jolicoeur-Martineau. 2018. The relativistic discriminator: a key element missing from standard GAN. *arXiv preprint arXiv:1807.00734* (2018).

[21] 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.

[22] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[23] H. Ko, D. Y. Lee, S. Cho, and A. C. Bovik. 2020. Quality Prediction on Deep Generative Images. *IEEE Transactions on Image Processing* 29 (2020), 5964–5979.

[24] Christian Ledig, Lucas Theis, Ferenc Huszar, Jose Caballero, Andrew P. Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, and Wenzhe Shi. 2016. Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network. *CoRR* abs/1609.04802 (2016).

[25] Cheng-Han Lee, Ziwei Liu, Lingyun Wu, and Ping Luo. 2020. MaskGAN: Towards Diverse and Interactive Facial Image Manipulation. In *Proc. of CVPR.*

[26] Vitaliy Lyudvichenko, Mikhail Erofeev, Yury Gitman, and Dmitriy Vatolin. 2017. A semiautomatic saliency model and its application to video compression. In *Proc. of IEEE International Conference on Intelligent Computer Communication and Processing (ICCP).*

[27] Vitaliy Lyudvichenko, Mikhail Erofeev, Alexander Ploshkin, and Dmitriy Vatolin. 2019. Improving Video Compression with Deep Visual-Attention Models. In *Proc. of International Conference on Intelligent Medicine and Image Processing* (Bali, Indonesia) *(IMIP '19).* Association for Computing Machinery, New York, NY, USA, 88–94. https://doi.org/10.1145/3332340.3332358

[28] Danial Maleki, Soheila Nadalian, Mohammad Mahdi Derakhshani, and Mohammad Amin Sadeghi. 2018. BlockCNN: A Deep Network for Artifact Removal and Image Compression.. In *CVPR Workshops.* 2555–2558.

[29] 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.*

[30] I. Mitrica, A. Fiandrotti, M. Cagnazzo, E. Mercier, and C. Ruellan. 2019. Cockpit Video Coding with Temporal Prediction. In *Proc. of EUVIP.* 28–33.

[31] A. Mittal, A. K. Moorthy, and A. C. Bovik. 2012. No-Reference Image Quality Assessment in the Spatial Domain. *IEEE Transactions on Image Processing* 21, 12 (Dec 2012), 4695–4708.

[32] 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.

[33] Anish Mittal, Rajiv Soundararajan, and Alan C Bovik. 2013. Making a "completely blind" image quality analyzer. *IEEE Signal Processing Letters* 20, 3 (2013), 209–212.

[34] Omkar M Parkhi, Andrea Vedaldi, and Andrew Zisserman. 2015. Deep face recognition. (2015).

[35] Alessandro Redondi, Luca Baroffio, Lucio Bianchi, Matteo Cesana, and Marco Tagliasacchi. 2016. Compress-then-analyze versus analyze-then-compress: What is best in visual sensor networks? *IEEE Transactions on Mobile Computing* 15, 12 (2016), 3000–3013.

[36] Oren Rippel and Lubomir Bourdev. 2017. Real-time adaptive image compression. In *Proc. of ICML.*

[37] Oren Rippel, Sanjay Nair, Carissa Lew, Steve Branson, Alexander G Anderson, and Lubomir Bourdev. 2019. Learned video compression. In *Proc. of ICCV.* 3454–3463.

[38] Andreas Rössler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. 2019. FaceForensics++: Learning to Detect Manipulated Facial Images. In *Proc. of International Conference on Computer Vision.*

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

[40] Hossein Talebi, Damien Kelly, Xiyang Luo, Ignacio Garcia Dorado, Feng Yang, Peyman Milanfar, and Michael Elad. 2020. Better Compression with Deep Pre-Editing. *arXiv preprint arXiv:2002.00113* (2020).

[41] Xiaoli Wang, Aakanksha Chowdhery, and Mung Chiang. 2016. SkyEyes: Adaptive Video Streaming from UAVs. In *Proc.of Workshop on Hot Topics in Wireless* (New York City, New York) *(HotWireless '16).* Association for Computing Machinery, New York, NY, USA, 2–6. https://doi.org/10.1145/2980115.2980119

[42] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. 2018. Esrgan: Enhanced super-resolution generative adversarial networks. In *Proceedings of the European Conference on Computer Vision (ECCV).* 0–0.

[43] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing* 13, 4 (2004), 600–612.

[44] 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 IEEE Computer Vision and Pattern Recognition.*

[45] Maarten Wijnants, Sven Coppers, Gustavo Rovelo Ruiz, Peter Quax, and Wim Lamotte. 2019. Talking Video Heads: Saving Streaming Bitrate by Adaptively Applying Object-Based Video Principles to Interview-like Footage. In *Proc. of ACM Multimedia* (Nice, France) *(MM '19).* Association for Computing Machinery, New York, NY, USA, 2449–2458. https://doi.org/10.1145/3343031.3351045

[46] Jaeyoung Yoo, Sang-ho Lee, and Nojun Kwak. 2018. Image Restoration by Estimating Frequency Distribution of Local Patches. In *Proc. of IEEE Computer Vision and Pattern Recognition.*

[47] Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, and Nong Sang. 2018. Bisenet: Bilateral segmentation network for real-time semantic segmentation. In *Proc. of European Conference on Computer Vision.* 325–341.

[48] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. 2018. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In *CVPR.*