Extreme Learned Image Compression with GANs

Eirikur Agustsson* Michael Tschannen* Fabian Mentzer* Radu Timofte Luc Van Gool
aeirikur@vision.ee.ethz.ch michaelt@nari.ee.ethz.ch mentzerf@vision.ee.ethz.ch timofter@vision.ee.ethz.ch vangool@vision.ee.ethz.ch
ETH Zürich, Switzerland

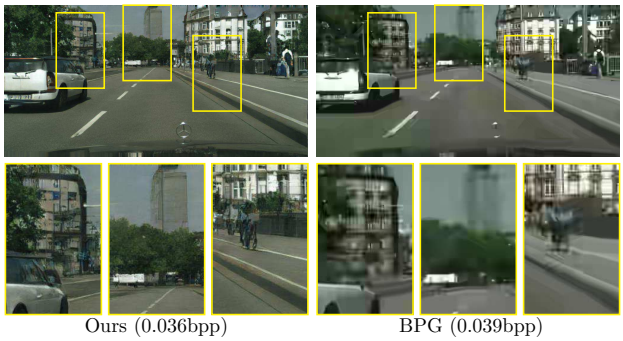


Figure 1: Images produced by our *global generative compression* network trained with an adversarial loss, along with the corresponding results for BPG.

Abstract

We propose a framework for extreme learned image compression based on Generative Adversarial Networks (GANs), obtaining visually pleasing images at significantly lower bitrates than previous methods. This is made possible through our GAN formulation of learned compression combined with a generator/decoder which operates on the full-resolution image and is trained in combination with a multi-scale discriminator. Additionally, our method can fully synthesize unimportant regions in the decoded image such as streets and trees from a semantic label map extracted from the original image, therefore only requiring the storage of the preserved region and the semantic label map. A user study confirms that for low bitrates, our approach significantly outperforms state-of-the-art methods, saving up to 67% compared to the next-best method BPG.

1. Introduction

Image compression systems based on deep neural networks (DNNs), or deep compression systems for short, have become an active area of research recently. These systems often outperform state-of-the-art engineered codecs such as BPG, WebP, and JPEG2000 on perceptual metrics [13, 3, 11, 4, 10]. Besides achieving higher compression rates on natural images, they can be easily adapted to specific target domains such as stereo or medical images, and promise efficient processing and indexing directly from compressed representations [16]. However, for bitrates below 0.1 bits per pixel (bpp) these algorithms still incur severe quality reductions and common training objectives such as peak signal-to-noise ratio (PSNR) or multi-scale structural similarity (MS-SSIM) become meaningless as they favor exact preservation of local (high-entropy) structure over preserving texture. To further advance deep image compression it is therefore of great importance to develop new training objectives beyond PSNR and MS-SSIM. A promising candidate towards this goal are adversarial losses [6] which were shown recently to capture global semantic information and local texture, yielding powerful generators that produce visually appealing high resolution images from semantic label maps [7, 17].

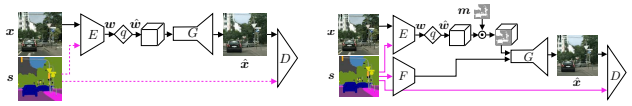
In this paper, we propose and study a generative adversarial network (GAN)-based framework for extreme image compression, targeting bitrates below 0.1 bpp. We present a principled GAN formulation for deep image compression that allows for different degrees of content generation. In contrast to prior works on deep image compression which applied adversarial losses to image patches for artifact suppression [11, 5] and generation of texture details [9] or representation learning for thumbnail images [12], our generator/decoder operates on the full-resolution image and is trained with a multi-scale discriminator [17].

We study two modes of operation (corresponding to unconditional and conditional GANs), namely

- *global generative compression (GC)*, preserving the overall image content while generating structure of different scales such as leaves of a tree or windows in the facade of buildings, and
- *selective generative compression (SC)*, completely generating parts of the image from a semantic label map while preserving user-defined regions with a high degree of detail.

A typical use case for GC are bandwidth constrained sce-

*Equal contribution.



(a) Global generative compression (GC)

(b) Selective generative compression (SC)

Figure 2: Structure of the proposed compression networks. E is the encoder for the image $\boldsymbol{x}$ and optionally the semantic label map $\mathbf{s}$. q quantizes the latent code $\boldsymbol{w}$ to $\hat{\boldsymbol{w}}$. G is the generator, producing the decompressed image $\hat{\boldsymbol{x}}$, and D the discriminator used for adversarial training. For SC, F extracts features from $\mathbf{s}$ and the subsampled heatmap multiplies $\hat{\boldsymbol{z}}$ (pointwise) for spatial bit allocation.

narios, where one wants to preserve the full image as much as possible, while falling back to synthesized content instead of blocky/blurry blobs for regions where there are not sufficient bits to store the original pixels. SC could be applied in a video call scenario where one wants to fully preserve people in the video stream, but a visually pleasing synthesized background serves our purpose as well as the true background. In the GC operation mode the image is transformed into a bitstream and encoded using arithmetic coding. SC requires a semantic/instance label map of the original image which can be obtained using off-the-shelf semantic/instance segmentation networks, and which is stored as a vector graphic. This amounts to a small, image dimension independent overhead in terms of coding cost. On the other hand, the size of the compressed image is reduced proportionally to the area which is generated from the semantic label map, typically leading to a significant overall reduction in storage cost.

Due to space limitations, we focus on GC in the present paper and refer to the full version [2] for a description of SC and a corresponding evaluation. Here, we present a comprehensive user study showing that our GC compression system yields visually considerably more appealing results than BPG (the current state-of-the-art engineered compression algorithm) and the recently proposed autoencoder-based deep compression (AEDC) system [10]. In particular, for the street scene images from the Cityscapes data set, users prefer the images produced by our method over BPG even when BPG uses more than double the bits. To the best of our knowledge, these are the first results showing that a deep compression method outperforms BPG in a user study.

Related work The most popular DNN architectures for image compression are to date auto-encoders [13, 3, 1, 16] and recurrent neural networks (RNNs) [14, 15].

Generative adversarial networks (GANs) [6] have emerged as a popular technique for learning generative models for intractable distributions in an unsupervised manner. [12] uses a GAN framework to learn a generative model over thumbnail images, which is then used as a decoder for thumbnail image compression. Other works use adversarial training for compression artifact removal [11, 5] and single image super-resolution [9].

2. GANs for extreme image compression

Our proposed GANs for extreme image compression can be viewed as a combination of (conditional) GANs and learned compression. With an encoder E and quantizer q, we encode the image $\boldsymbol{x}$ to a compressed representation $\hat{\boldsymbol{w}} = q(E(\boldsymbol{x}))$. This representation is optionally concatenated with noise $\boldsymbol{v}$ drawn from a fixed prior $p_{\boldsymbol{v}}$, to form the latent vector $\boldsymbol{z}$. The decoder/generator G then tries to generate an image $\hat{\boldsymbol{x}} = G(\boldsymbol{z})$ that is consistent with the image distribution $p_{\boldsymbol{x}}$ while also recovering the specific encoded image $\boldsymbol{x}$ to a certain degree (see Fig. 2 (a)). Using $\boldsymbol{z} = [\hat{\boldsymbol{w}}, \boldsymbol{v}]$, this can be expressed by our saddle-point objective for (non-conditional) generative compression,

$$\min_{E,G} \max_{D} \quad \mathbb{E}[f(D(\boldsymbol{x}))] + \mathbb{E}[g(D(G(\boldsymbol{z}))] + \lambda \mathbb{E}[d(\boldsymbol{x}, G(\boldsymbol{z}))] + \beta H(\hat{\boldsymbol{w}}), \qquad (1)$$

where $\lambda > 0$ balances the distortion term against the GAN loss and entropy terms. Using this formulation, we need to encode a real image, $\hat{\boldsymbol{w}} = E(\boldsymbol{x})$, to be able to sample from $p_{\hat{\boldsymbol{w}}}$. However, this is not a limitation as our goal is to compress real images and not to generate completely new ones.

We note that equation (1) has completely different dynamics than a normal GAN, because the latent space $\boldsymbol{z}$ contains $\hat{\boldsymbol{w}}$, which stores information about a real image $\boldsymbol{x}$. A crucial ingredient is the bitrate limitation on $H(\hat{\boldsymbol{w}})$. If we allow $\hat{\boldsymbol{w}}$ to contain arbitrarily many bits by setting $\beta = 0$ and letting L and $\dim(\hat{\boldsymbol{w}})$ be large enough, E and G could learn to near-losslessly recover $\boldsymbol{x}$ from $G(\boldsymbol{z}) = G(q(E(\boldsymbol{x})))$, such that the distortion term would vanish. In this case, the divergence between $p_{\boldsymbol{x}}$ and $p_{G(\boldsymbol{z})}$ would also vanish and the GAN loss would have no effect.

By constraining the entropy of $\hat{\boldsymbol{w}}$, E and G will never be able to make d fully vanish. In this case, E, G need to balance the GAN objective $\mathcal{L}_{\text{GAN}}$ and the distortion term $\lambda \mathbb{E}[d(\boldsymbol{x}, G(\boldsymbol{z}))]$, which leads to $G(\boldsymbol{z})$ on one hand looking "realistic", and on the other hand preserving the original image. For example, if there is a tree for which E cannot afford to store the exact texture (and make d small) G can synthesize it to satisfy $\mathcal{L}_{\text{GAN}}$, instead of showing a blurry green blob.

3. Experiments

We only present the most important aspects; please see [2] for a full description.

Network architecture: The architecture for our encoder E and generator G is based on the global generator network

proposed in [17], which in turn is based on the architecture of [8]. For the discriminator D we use the multi-scale architecture of [17], which measures the divergence between $p_{\boldsymbol{x}}$ and $p_{G(\boldsymbol{z})}$ both locally and globally. For the distortion term we adopt $d(\boldsymbol{x}, \hat{\boldsymbol{x}}) = \text{MSE}$ and the feature matching and VGG perceptual losses as proposed in [17].

Datasets: We train the proposed method on two popular data sets that come with hand-annotated semantic label maps, namely *Cityscapes* and *ADE20k*. Both of these data sets were previously used with GANs [7], hence we know that GANs can model their distribution—at least to a certain extent. To assess how well our models generalize, we test the GC model with $C = 4$ on Kodak .

Training: We employ the ADAM optimizer with a learning rate of 0.0002 and set the mini-batch size to 1. Our networks are trained for 50 epochs on Cityscapes and for 20 epochs on ADE20k, aside from the network tested on Kodak which was trained for 50 epochs on ADE20k.

Baselines: We compare our method to the HEVC-based image compression algorithm BPG (in the default 4:2:2 chroma format) and to the AEDC network [10], trained for an operating point of 0.07 bpp (which obtains a slightly higher MS-SSIM than BPG at the same bpp).

User study: Quality measures such as PSNR and MS-SSIM commonly used to assess the quality of compression systems become meaningless at very low bitrates as they penalize changes in local structure rather than preservation of the global content. Therefore, to quantitatively evaluate the perceptual quality of our GC networks in comparison with BPG and AEDC we conduct a user study using the Amazon Mechanical Turk (AMT) platform.

For each pairing of methods on Cityscapes and ADE20K, we compare the decompressed images obtained for a set of 20 randomly picked validation images at different bpp, having as reference the downscaled 1024×512px images. For each pairing on Kodak, we used all 24 images of the dataset. 9 randomly selected users were asked to select the best decompression result for each test image and pairing of methods.

See [2] for quantitative evaluation of how well the different networks preserve the image semantics.

3.1. Results

In Tables 1 and 2 we report the percentages of preference of the image produced by the proposed method over the image produced by the other compression method for Cityscapes and ADE20k, respectively. For each method vs. method comparison 180 human opinions were collected. For both data sets, the perceptual quality of our results is better than that of the baseline approaches at comparable bpp. For Cityscapes, at 0.036 bpp our method is picked by the users over BPG in 81.87% of the cases, while at 0.072 bpp our method is preferred over BPG and AEDC in 70.18% and 84.21% of the cases, respectively.

In Fig. 1 we present example validation images from Cityscapes produced by our GC networks at different bpp along with the images obtained from the baseline algorithms at the same bpp. The GC produces images with finer structure than BPG, which suffers from smoothed patches and blocking artifacts. AEDC and our network trained for MSE both produce blurry images.

Preference of **our** results [%] vs.	BPG 0.039 bpp	BPG 0.056 bpp	BPG 0.072 bpp	BPG 0.079 bpp	BPG 0.1 bpp	AEDC [10] 0.069 bpp
our $C = 2$, 0.018 bpp	**76.02**	**52.05**	45.03	38.01	29.24	**71.93**
our $C = 4$, 0.036 bpp	**81.87**	**67.25**	**59.65**	**50.88**	35.67	**80.12**
our $C = 8$, 0.072 bpp	**83.63**	**74.27**	**70.18**	**67.84**	**50.88**	**84.21**

Table 1: User study quantitative preferences results [%] on Cityscapes. For each pairing of methods we report the percentage of cases in which the image produced by our GC method was preferred by human subjects over the result of the other compression method.

Preference of **our** results [%] vs.	BPG 0.054 bpp	BPG 0.064 bpp	BPG 0.072 bpp	BPG 0.082 bpp	BPG 0.1 bpp
our $C = 4$, 0.036 bpp	**66.67**	**52.63**	36.26	/	/
our $C = 8$, 0.072 bpp, w. sem.	**80.12**	**73.68**	**57.31**	**52.63**	41.52

Table 2: User study quantitative preferences results [%] on ADE20k. For comparable bpp our method is clearly preferred.

Preference of **our** results [%] vs.	BPG 0.038 bpp	BPG 0.060 bpp	BPG 0.065 bpp	BPG 0.072 bpp
our $C = 4$, 0.036 bpp	**87.10**	**57.60**	**54.84**	47.00

Table 3: User study quantitative preferences results [%] on Kodak. Our method is preferred over BPG at 0.065bpp, which corresponds to a 45% bitrate reduction.

Generalization to Kodak: We show the results for an example Kodak image in Figure 3, obtained with a model trained on ADE20K for GC (without semantics) using $C = 4$ channels (0.036 bpp). While there is some color shift noticeable (which could be accounted for by reducing the domain mismatch and/or increasing the weight of the perceptual loss), we see that our method can realistically synthesize details where BPG fails.

The user study results in Table 3 shows that our method is preferred over BPG, even when BPG uses an 80% larger bitrate of 0.065 bpp compared to our method at 0.036 bpp.

4. Discussion

Qualitatively, our GC networks preserve more and sharper structure than the baseline methods, for both the Cityscapes and ADE20k images. For both data sets, the user study shows that at a given target bpp humans on average prefer the pictures produced by our GC networks over


Kodak Image 13

Ours (0.036bpp)


BPG (0.073bpp)


JPEG2000 (0.037bpp)


WebP (0.078bpp)


JPEG (0.248bpp)

Figure 3: Original Kodak Image 13 along with the decompressed version used in the user study (Ours), obtained using our GC network.

BPG. For Cityscapes, where we trained an AEDC model, our images are on average also preferred over AEDC. The Cityscapes images obtained by our GC networks with $C = 2$ (0.018bpp) and $C = 4$ (0.036 bpp) were even preferred over BPG at 0.056 and BPG at 0.079 bpp, respectively, showing that our method outperforms BPG even when BPG uses more than twice as many bits. For ADE20k, the results produced by our GC networks were preferred on average by a considerable margin over BPG, although the preference is less pronounced than for Cityscapes.

Furthermore, we found that our model trained on ADE20K can also generalize well to the Kodak dataset, being preferred over BPG for $C = 4$ (0.036bpp) even when BPG uses 80% more bits.

We note that while prior works [11, 10, 4] have outperformed BPG in terms of MS-SSIM, they have not demonstrated improved visual quality over BPG (which is optimized for PSNR). In particular, [4, 10] show a visual comparison but do not claim improved visual quality over BPG, whereas [11] does not compare with BPG visually. To the best of our knowledge, this is the first time that a deep compression method is shown to outperform BPG in a user study—and that with a large margin.

References

[1] E. Agustsson, F. Mentzer, M. Tschannen, L. Cavigelli, R. Timofte, L. Benini, and L. Van Gool. Soft-to-hard vector quantization for end-to-end learning compressible representations. *arXiv preprint arXiv:1704.00648*, 2017. 2

[2] 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. 2, 3

[3] J. Ballé, V. Laparra, and E. P. Simoncelli. End-to-end optimized image compression. *arXiv preprint arXiv:1611.01704*, 2016. 1, 2

[4] J. Ballé, D. Minnen, S. Singh, S. J. Hwang, and N. Johnston. Variational image compression with a scale hyperprior. In *ICLR*, 2018. 1, 4

[5] L. Galteri, L. Seidenari, M. Bertini, and A. Del Bimbo. Deep generative adversarial compression artifact removal. In *CVPR*, pages 4826–4835, 2017. 1, 2

[6] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014. 1, 2

[7] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, pages 1125–1134, 2017. 1, 3

[8] J. Johnson, A. Alahi, and L. Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision*, 2016. 3

[9] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *CVPR*, pages 4681–4690, 2017. 1, 2

[10] F. Mentzer, E. Agustsson, M. Tschannen, R. Timofte, and L. Van Gool. Conditional probability models for deep image compression. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 1, 2, 3, 4

[11] O. Rippel and L. Bourdev. Real-time adaptive image compression. In *ICML*, volume 70, pages 2922–2930. PMLR, 06–11 Aug 2017. 1, 2, 4

[12] S. Santurkar, D. Budden, and N. Shavit. Generative compression. *arXiv preprint arXiv:1703.01467*, 2017. 1, 2

[13] L. Theis, W. Shi, A. Cunningham, and F. Huszar. Lossy image compression with compressive autoencoders. In *International Conference on Learning Representations (ICLR)*, 2017. 1, 2

[14] G. Toderici, S. M. O'Malley, S. J. Hwang, D. Vincent, D. Minnen, S. Baluja, M. Covell, and R. Sukthankar. Variable rate image compression with recurrent neural networks. *arXiv preprint arXiv:1511.06085*, 2015. 2

[15] G. Toderici, D. Vincent, N. Johnston, S. J. Hwang, D. Minnen, J. Shor, and M. Covell. Full resolution image compression with recurrent neural networks. *arXiv preprint arXiv:1608.05148*, 2016. 2

[16] R. Torfason, F. Mentzer, E. Ágústsson, M. Tschannen, R. Timofte, and L. V. Gool. Towards image understanding from deep compression without decoding. In *ICLR*, 2018. 1, 2

[17] 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. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 1, 3, 5