User-Guided Variable Rate Learned Image Compression

Rushil Gupta[1*], Suryateja BV[1], Nikhil Kapoor[2*], Rajat Jaiswal[2*], Sharmila Nangi[3], Kuldeep Kulkarni[1]
[1]Adobe Research, Bengaluru, India
[2]Indian Institute of Technology Delhi, India
[3]Stanford University, USA
{rusgupta, surbv, kulkulka}@adobe.com {mt1170739, cs5170415}@iitd.ac.in srnangi@stanford.edu

Abstract

We propose a learning-based image compression method that achieves any arbitrary input bitrate via user-guided bit allocation to preferred regions. We verify our hypothesis of incorporating user guidance for bitrate control by experimenting with alternatives that do not have any guidance. We conduct extensive evaluation on CelebA-HQ and CityScapes dataset using standard quantitative metrics and human studies showing that our single model for multiple bitrates achieves similar or better performance as compared to previous learned image compression methods that require re-training for each new bitrate.

1. Introduction

A desirable feature of any image compression algorithm is the flexibility to achieve a bitrate specified by users as they have strict storage budget requirements. Traditional codecs like JPEG [29], JPEG2000 [25] or HEVC [24] partially allow users to control bitrate through quality factors. However, deep learning-based algorithms [2–5, 17, 18, 22, 26–28] suffer from the drawback that the network is tightly coupled to a single bitrate, as governed by the weight in rate-distortion trade-off [7, 23] term used while training. Thus, multiple networks have to be trained to achieve different bitrates, which is both costly and time-intensive. A few attempts [8, 10, 12, 28] have been made to address this drawback by training a single network for variable bitrates. None of these approaches provide either theoretical or empirical guarantees of achieving the desired (user-input) bitrate during test time and generally, the output bitrate is controlled by fine-tuning a few proxy parameters. Hence users are burdened with the task of having to second-guess the values that map to the exact bitrate that is desired and thus might have to perform several forward passes through the network till the desired rate is achieved.

We present a novel approach to user-guided image compression that enables users to have direct control over bitrate. The core idea of our method is to allow the user to provide a relative importance map as input to the network, which is of the same size as the input image wherein the user specifies a certain importance value to every region in the image. We hypothesize that this form of user guidance enables optimal tracking of desired bitrate. Work by [15] attempts object-adaptive image compression by learning an importance map in an unsupervised manner. The learned map is quantized and encoded in the form of bits, which enables direct control of the final bitrate achieved. However, since the importance map itself is a learned output of one of the layers of the network as in [17], the user has no control of either the desired bitrate or which regions to allocate more bits to.

The contributions of this paper are as follows:

- We propose a novel GAN-based and user-guided image compression algorithm that allows users to input desired bitrates. The generator is guided by a user-provided relative importance map that is used to achieve the desired bitrate.
- We propose a novel loss function, called the equivalence distortion (ED) loss, that constrains the learned importance map to be region-wise close to the input importance map, thereby aiding to meet the bitrate constraint.
- We show through quantitative experiments and human studies for CelebA-HQ and Cityscapes dataset that a **single** model is able to achieve high quality image reconstructions for a wide-range of input bitrates. Further, we show our method performs nearly as well as traditional compression methods state-of-the-art learned image compression methods for a broad range of bitrates in terms of PSNR, MS-SSIM, FID and KID metrics.

2. Learning One Model for Multiple Bitrates

Our problem statement is as follows: Given an input image I, user-provided input bitrate t and relative importance val-

[*] This work was done during an internship at Adobe Research, Bengaluru

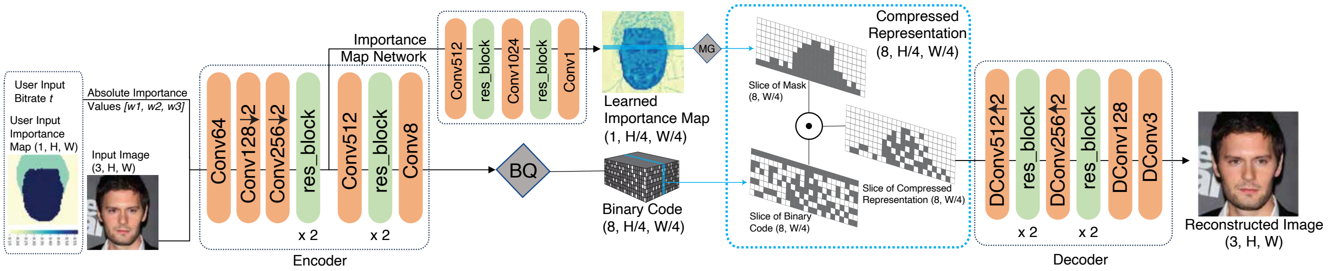


Figure 1. End-to-end pipeline of our compression model with $(C, n^2) = (8, 16)$. ConvC is a convolution with C channels, $\uparrow 2, \downarrow 2$ indicate strided up or down convolutions, BQ refers to Binary Quantizer. MG refers to Mask Generator. Method to obtain absolute importance values is described in Section 5 of supplementary material. Discriminator is used only during training.

ues $(p_i)_{i=1}^h$ for h different regions in image I, our goal is to train a **single model** that (1) learns an importance map m_l that allocates bits to h regions in the order of their importance, and (2) maintains an output bitrate t_{out} as close as possible to the user-input bitrate t.

To achieve this, we introduce our lossy compression method that is based on a GAN framework and follows an adversarial training procedure. Figure 1 shows the architecture of our compression method and an example of how importance values get distributed within a region (details in Figure 3 of supplementary). In the following sections, we describe our architecture and the loss functions used to train our model.

2.1. Notations

We denote the user-input bitrate by t, user-input relative importance map as m_r, absolute importance map as m_a and the input image as I. The segmentation map for any region i of the image I is denoted by s_i. The latent representation generated by encoder is denoted by z and the binary code by q. Importance Map Network gives a learned importance map denoted by m_l and the Mask Generator (MG) converts m_l into m_q. Further, the compressed representation is denoted as c_I and the reconstructed image is represented by $\tilde{I}$. C, n respectively represent the number of channels in the latent representation z and the down-sampling factor of encoder. All the primed notations are the nearest-neighbour down-sampled versions of their base notations.

2.2. Architecture

Encoder: generates a latent representation z of shape $(C, H/n, W/n)$ from the concatenation of absolute importance map m_a and input image of shape $(3 + 1, H, W)$. It comprises of a series of convolution layers and residual blocks [26]. In our experiments, we set $(C, n) = (8, 4)$.

Importance Map Network: takes an intermediate input representation from the encoder and generates a single-channel learned importance map m_l that is close to the absolute importance map m_a. Each pixel of this learned map contains values between 0 and 1, dictating the number of latent representation channels to use for storing the information of that pixel. Architecture is inspired from [15].

Mask Generator: takes the learned importance map m_l (1, H/4, W/4) and quantizes it into a mask m_q (C, H/4, W/4) that essentially dictates the number of channels to be used to store information at each point in the encoded representation. Adapted from [15], $(m_q)_{kij}$ is given by 1 if $k < C * m_l(i, j)$ and 0 otherwise.

Binary Quantizer: binarizes z by converting each value to either -1 or 1, following the work of [27]. This quantized output q is used to get c_I by taking a Hadamard Product with the quantized mask, m_q generated from learned importance map m_l. Since there is a loss of information at this step, our compression algorithm is lossy.

Decoder: mirrors the encoder with a series of deconvolution layers and residual blocks. It reconstructs the image, $\tilde{I}$ from the compressed latent representation c_I of input image I. In our experiments, all models use transposed convolution for up-sampling.

Discriminator: plays a key role in ensuring photo-realism of reconstructed images as it attempts to distinguish between input (original) image I and reconstructed (fake) image $\tilde{I}$. It takes in both the images concatenated with segmentation maps s_i and learned importance map m_l as its input. We use a multi-scale version [20] consisting of four discriminators, each operating at a different image scale and having five convolution layers each.

2.3. Equivalence Distortion (ED) Loss Function

We introduce a novel loss function to ensure bits are allocated optimally in the importance map while staying within the limits of user-provided bit budget. It comprises of two terms: Distortion Loss (L_D) and Equivalence Loss (L_E). L_D computes MSE Loss and MS-SSIM between the input and reconstructed images and thus, affects the reconstruction ability of the model. L_E affects the bitrate allocation of the model and comprises of three terms: L_{whole} that penalizes the model when the sum of values in learned map exceed the user input map, and two region-wise terms, L^1_{region} and L^2_{region} that compare the maps region-wise and enforce the model to adaptively distribute importance values within a region in the learned map by giving higher weightage to areas with high-texture or edges and lower weightage to flat areas. The total loss, L_{ED} then becomes $L_{ED} = L_E + L_D$. Additional losses are described in the supplementary.

3. Experiments and Results

3.1. Setup

Datasets: We use CelebAMask-HQ faces dataset [13, 14, 16] and CityScapes dataset [9] to train and evaluate our proposed method. CelebAMask-HQ contains 30,000 high resolution (1024x1024) face images each having a 512x512 segmentation map with 19 classes. We club the classes into 3 broad classes, i.e. Face, Hair, and Background. All the images are resized to 256x256 dimensions. For the CityScapes dataset, we club the object classes and consider broadly 5 classes, viz. humans, vehicle, object, construction and others (details in supplementary). All the images are resized to 512x1024 dimensions.

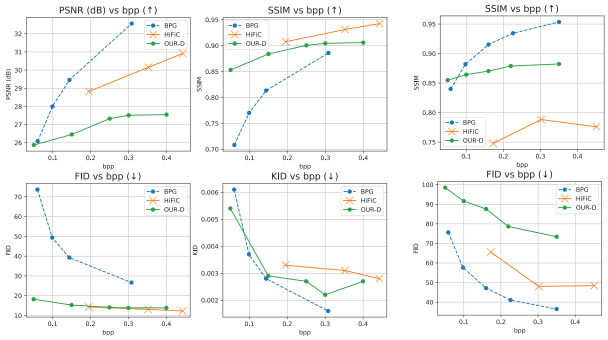
Baselines: We compare our proposed method with BPG and High-Fidelity generative Compression (HiFiC) [18]. We consider HiFiC with GAN component as the closest baseline to our work [1, 19]. Note that HiFiC has to be trained three times on each dataset to obtain the -low, -med, and -high variants of bitrates, unlike our method which has to be trained *only once*. Additionally, there is no notion of region-wise user input importance in the baselines, which is a unique feature of our method.

Evaluation Metrics: We measure the quality of our compression method using standard pixel-wise metrics like PSNR and MS-SSIM, and perceptual metrics like FID [11] and KID [6] scores. All metrics are computed on the entire image except for region-wise PSNR scores that depend on a particular region. We experimented with various values of C and n that provide a maximum bitrate (C/n^2) of 0.5 and empirically found ($C = 8, n^2 = 16$) to work the best, with its corresponding results are discussed below.

3.2. Results

Gains from User Guidance We compare our method of controlling output bitrate (t_{out}) via user-guided relative importance values with two alternatives that **do not** have any user guidance/segmentation masks. The results and analysis of the comparison is included in supplementary sec 8.2.

Varying Bitrates and Importance Values. In Figure 2, we present the trends of perceptual metrics for our method and baselines as we vary bitrates. Further, we present qualitative examples of reconstructed images and their learned importance maps as we vary user-provided relative importance values. We increase the importance of one class (face for CelebA, construction for CityScapes) and decrease the importance of another by the same amount, keeping remaining classes at same/equal importance. Figures 4 & 5 validate the utility of our approach in incorporating region-wise user preferences.



(a) CelebA

(b) CityScapes

Figure 2. Comparison of our proposed method **Our-D** on (a) CelebA; (b) CityScapes against baselines (BPG, HiFiC) as the bitrate is varied keeping the importances of all classes equal. We see that **Our-D** is close to what is achieved by the baselines in most metrics. This validates the efficacy of our approach of training a single model for wide range of bitrates while performing on par with baselines that are trained for a specific bitrate.

Human Evaluation. Since our method is a user-facing technology, we conduct extensive human studies to gauge the perceived visual quality of reconstructed images [21]. We use Amazon Mechanical Turk (AMT) for crowdsourcing annotations through two surveys that compare images generated by different baselines and our method. For quality control, we set the annotator prerequisites to ”MTurk Masters” having an approval rate more than 95% in at least past 30 annotations. We solicit five unique responses for each datapoint. We conducted a pilot study and asked for textual feedback from annotators to improve our questions. From Figure 3, we observe that our model may not outperform baselines on quantitative metrics but it performs better for all cases in human evaluation. While annotators prefer HiFiC for CityScapes overall reconstruction, they prefer our model when asked for region-wise (vechile) quality. When asked to rate the overall quality, they might have not noticed the finer details in diverse CityScapes images, thus preferring HiFiC which is trained for that particular bitrate and dataset. However when asked to look carefully at the vehicle region, our model fares better.

4. Limitations and Future Work

Our method requires semantic label maps to train, and we experiment on domain-specific datasets like CelebA-HQ and Cityscapes wherein the exhaustive set of object classes is known to us. In future work, we intend to learn classes and semantic maps within our method. We also intend to reduce the gap between the input and output bitrates, and ensure good performance at higher bitrates without saturation, which can be potentially achieved with the addition of an entropy coding scheme.

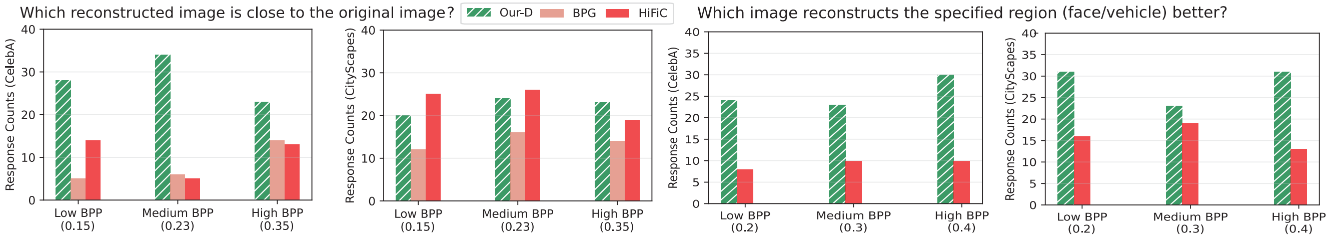


Figure 3. Results of Human Study across both datasets. From the first survey (left), we find that users find reconstructions from our method to be closer to original image. From the second survey (right), we find that our method with *high* relative importance for specified region is preferred by users as compared to HiFiC at same bitrate.

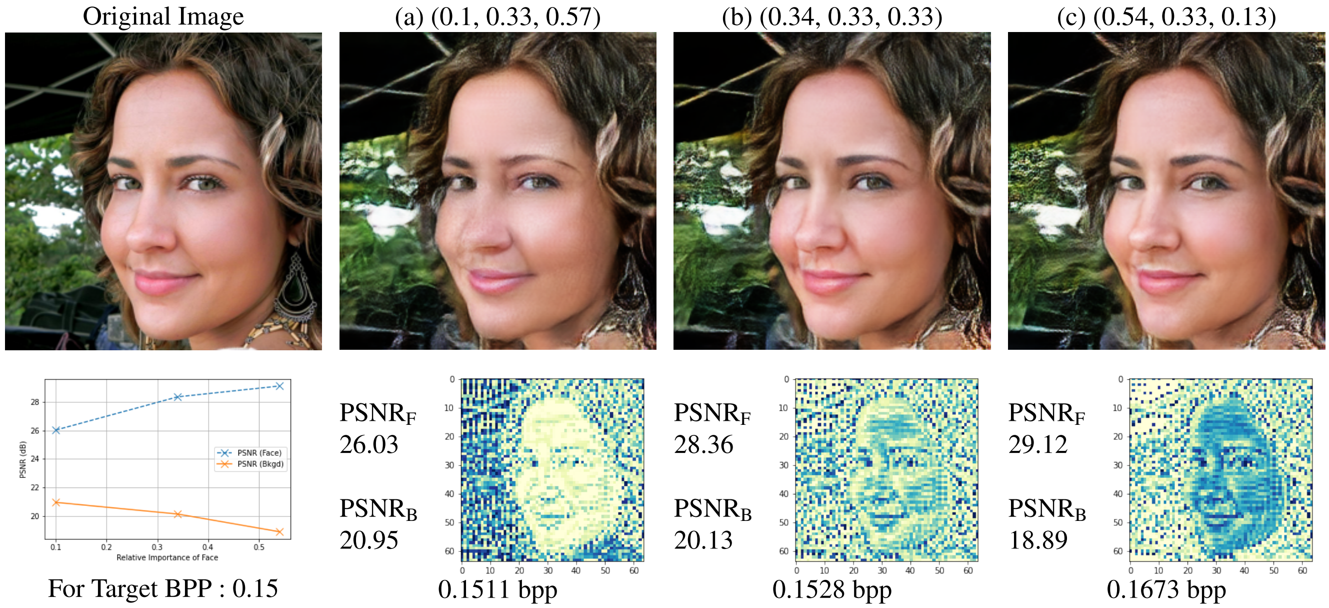


Figure 4. Comparison of reconstructed images and learned importance maps from CelebA dataset with different user-provided importances. The region-wise PSNR are also shown. The achieved bpp is nearly the same as the input target bpp even as we vary improtances. The number of bits allocated to face in learned maps increase leading to improved reconstruction quality of face. Consequently, the degradation in quality of background is evident.

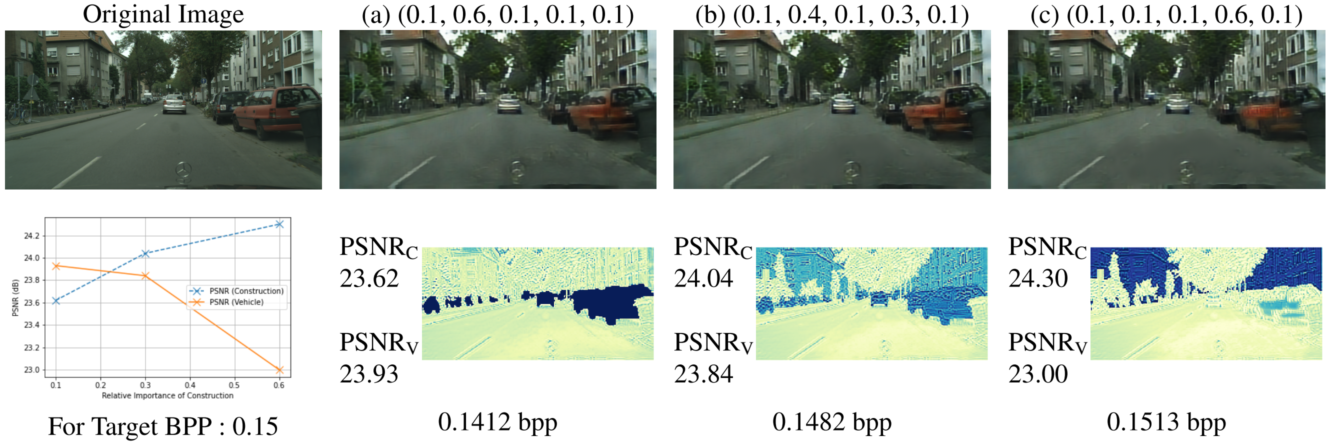


Figure 5. We can see that the red car on the right side gets progressively smudgy, while the buildings and windows get sharper. We thus validate our hypothesis on CityScapes dataset which is relatively more complex than the CelebA dataset both in terms of the variety in objects and the number of different classes of objects.

References

[1] Implementation of HiFiC using Pytorch. https://github.com/Justin-Tan/high-fidelity-generative-compression. 3

[2] Eirikur Agustsson, Fabian Mentzer, Michael Tschannen, Lukas Cavigelli, Radu Timofte, Luca Benini, and Luc Van Gool. Soft-to-hard vector quantization for end-to-end learning compressible representations, 2017. 1

[3] Eirikur Agustsson, Michael Tschannen, Fabian Mentzer, Radu Timofte, and Luc Van Gool. Generative adversarial networks for extreme learned image compression. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 221–231, 2019. 1

[4] Johannes Ballé, Valero Laparra, and Eero P Simoncelli. End-to-end optimized image compression. *arXiv preprint arXiv:1611.01704*, 2016. 1

[5] Johannes Ballé, David Minnen, Saurabh Singh, Sung Jin Hwang, and Nick Johnston. Variational image compression with a scale hyperprior. *arXiv preprint arXiv:1802.01436*, 2018. 1

[6] Mikolaj Binkowski, Danica J. Sutherland, Michael Arbel, and Arthur Gretton. Demystifying MMD gans. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. 3

[7] Yochai Blau and Tomer Michaeli. Rethinking lossy compression: The rate-distortion-perception tradeoff. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 675–685. PMLR, 2019. 1

[8] Yoojin Choi, Mostafa El-Khamy, and Jungwon Lee. Variable rate deep image compression with a conditional autoencoder. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3146–3154, 2019. 1

[9] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016. 3

[10] Emilien Dupont, Adam Goliński, Milad Alizadeh, Yee Whye Teh, and Arnaud Doucet. Coin: Compression with implicit neural representations, 2021. 1

[11] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6629–6640, Red Hook, NY, USA, 2017. Curran Associates Inc. 3

[12] Nick Johnston, Damien Vincent, David Minnen, Michele Covell, Saurabh Singh, Troy Chinen, Sung Jin Hwang, Joel Shor, and George Toderici. Improved lossy image compression with priming and spatially adaptive bit rates for recurrent networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4385–4393, 2018. 1

[13] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation, 2018. 3

[14] Cheng-Han Lee, Ziwei Liu, Lingyun Wu, and Ping Luo. Maskgan: Towards diverse and interactive facial image manipulation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 3

[15] Mu Li, Wangmeng Zuo, Shuhang Gu, Debin Zhao, and David Zhang. Learning convolutional networks for content-weighted image compression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3214–3223, 2018. 1, 2

[16] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild, 2015. 3

[17] Fabian Mentzer, Eirikur Agustsson, Michael Tschannen, Radu Timofte, and Luc Van Gool. Conditional probability models for deep image compression. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4394–4402, 2018. 1

[18] Fabian Mentzer, George Toderici, Michael Tschannen, and Eirikur Agustsson. High-fidelity generative image compression, 2020. 1, 3

[19] Fabian Mentzer, George D Toderici, Michael Tschannen, and Eirikur Agustsson. High-fidelity generative image compression. *Advances in Neural Information Processing Systems*, 33, 2020. 3

[20] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 2

[21] Yash Patel, Srikar Appalaraju, and R. Manmatha. Human perceptual evaluations for image compression, 2019. 3

[22] Yash Patel, Srikar Appalaraju, and R. Manmatha. Saliency driven perceptual image compression. *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 227–236, 2021. 1

[23] C. E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27(3):379–423, 1948. 1

[24] G. J. Sullivan, J. Ohm, W. Han, and T. Wiegand. Overview of the high efficiency video coding (hevc) standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 22(12):1649–1668, 2012. 1

[25] David Taubman and Michael Marcellin. *JPEG2000 Image Compression Fundamentals, Standards and Practice*. Springer Publishing Company, Incorporated, 2013. 1

[26] Lucas Theis, Wenzhe Shi, Andrew Cunningham, and Ferenc Huszár. Lossy image compression with compressive autoencoders. *arXiv preprint arXiv:1703.00395*, 2017. 1, 2

[27] George Toderici, Sean M O'Malley, Sung Jin Hwang, Damien Vincent, David Minnen, Shumeet Baluja, Michele Covell, and Rahul Sukthankar. Variable rate image compression with recurrent neural networks. *arXiv preprint arXiv:1511.06085*, 2015. 1, 2

[28] George Toderici, Damien Vincent, Nick Johnston, Sung Jin Hwang, David Minnen, Joel Shor, and Michele Covell.

Full resolution image compression with recurrent neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5306–5314, 2017. 1

[29] G. K. Wallace. The jpeg still picture compression standard. *IEEE Transactions on Consumer Electronics*, 38(1):xviii–xxxiv, 1992. 1