Adaptive Bitrate Quantization Scheme Without Codebook for Learned Image Compression

Jonas Löhdefink Jonas Sitzmann Andreas Bär Tim Fingscheidt
Technische Universität Braunschweig, Germany
{j.loehdefink, j.sitzmann, andreas.baer, t.fingscheidt}@tu-bs.de

Abstract

We propose a generic approach to quantization without codebook in learned image compression called one-hot max (OHM, Ω) quantization. It reorganizes the feature space resulting in an additional dimension, along which vector quantization yields one-hot vectors by comparing activations. Furthermore, we show how to integrate Ω quantization into a compression system with bitrate adaptation, i.e., *full control over bitrate during inference. We perform experiments on both MNIST and Kodak and report on rate-distortion trade-offs comparing with the integer rounding reference. For low bitrates ($<$ 0.4 bpp), our proposed quantizer yields better performance while exhibiting also other advantageous training and inference properties. Code is available at* `https://github.com/ifnspaml/OHMQ`.

1. Introduction

In learned image compression, autoencoders combined with quantizers and entropy models serve as the central building blocks [16]. The bitrate depends on the rate-distortion (RD) trade-off [28], typically being imprinted during training by the RD loss. While improvements have been achieved in the architectures of encoders, decoders, entropy models, and also loss functions, for quantization naive techniques are still widely applied.

A simple quantization approach is to round each bottleneck data point towards the next integer value element-wise — a special form of scalar quantization with a non-learned, uniform codebook. In the backward pass, the quantization error is simulated by additive uniformly distributed noise [5, 27]. Hence, during training, the autoencoder does not really adjust to the quantizer but merely learns to be error-tolerant.

To improve quantization, we propose a novel quantizer architecture depicted in Figure 1. Our proposed one-hot max (Ω) quantizer operates on reorganized input data with an additional dimension, and performs one-hot max vector quantization along this additional dimension. During backpropagation training, we approximate our Ω quantizer using the softmax function, following [1]. In contrast to other works on vector quantization [1, 14], our proposed Ω quantizer does not require an explicit codebook and enables flexible bitrate adaptation during inference with a single trained model, *i.e.*, one can manually specify the desired operating point on the RD curve at any point in time.

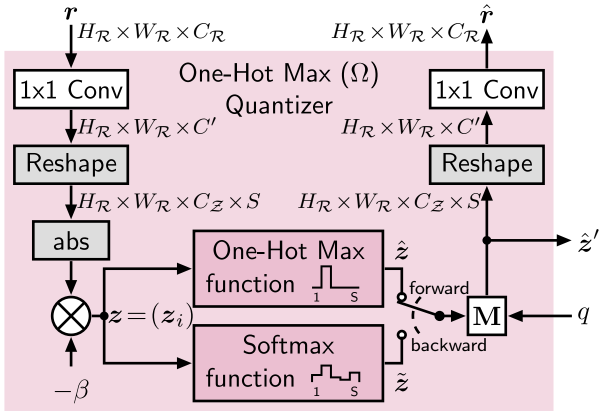

$\boldsymbol{r}$
$H_\mathcal{R} \times W_\mathcal{R} \times C_\mathcal{R}$
$H_\mathcal{R} \times W_\mathcal{R} \times C_\mathcal{R}$
$\hat{\boldsymbol{r}}$
1x1 Conv
One-Hot Max (Ω) Quantizer
1x1 Conv
$H_\mathcal{R} \times W_\mathcal{R} \times C'$
$H_\mathcal{R} \times W_\mathcal{R} \times C'$
Reshape
Reshape
$H_\mathcal{R} \times W_\mathcal{R} \times C_\mathcal{Z} \times S$
$H_\mathcal{R} \times W_\mathcal{R} \times C_\mathcal{Z} \times S$
abs
One-Hot Max function
1
S
$\hat{\boldsymbol{z}}$
$\hat{\boldsymbol{z}}'$
forward
$\boldsymbol{z} = (\boldsymbol{z}_i)$
M
q
backward
Softmax function
1
S
$\tilde{\boldsymbol{z}}$
$-\beta$

Figure 1. **One-hot max (Ω) quantizer** with one-hot output vectors $\hat{\boldsymbol{z}}'$ obtained by the one-hot max function. In the backward pass, the quantization is approximated by the differentiable softmax function. For learned image compression, the quantizer is embedded in a compression architecture as shown in Figure 2.

Our contributions are as follows: First, we propose the one-hot max (Ω) quantizer which does not rely on a codebook. Second, we propose a bitrate adaptation mechanism based on feature map masking with a special sampling protocol during training. This enables image compression at various bitrates with a single trained model. Third, we show a generic learned image compression system including encoder, decoder, entropy model, and our proposed Ω quantization. Fourth, we compare our proposed quantizer with an integer-rounding-based reference on MNIST and Kodak.

2. Related Work

In learned image compression, quantization, being discretization of continuous values [11–13, 29], is applied to neural network activations to yield compact data representations [5, 32]. Scalar quantization — *e.g.*, by rounding to the nearest integer [2, 6, 27] — and vector quantization [1, 12, 22, 24, 35] rival in terms of low complexity and low distortion. Both variants make use of explicit codebooks [12, 22, 24, 35]. More recent methods jointly optimize networks and codebooks such as VQ-VAE [33], where straight-through estimation (STE) [7] provides gradients to the encoder. Also, applying a softmax-based estimation with temperature has emerged [1, 14, 35, 37]. In this work, we adopt the softmax approximation, while in contrast to [1, 14], our method does not require a codebook. In our experimental evaluation, we compare our proposed Ω quantization to integer-rounding-based (INT) quantization as is used, *e.g.*, in [2, 6, 27]. Note that this work *does not aim* at some benchmark absolute compression system performance, *rather it concentrates on quantizer schemes.* In consequence, we embed our proposed Ω quantizer in a fairly simple autoencoder, being adopted from [27].

Concerning adaptive compression, two types of dynamic bit allocation are often applied. First, many approaches allow spatially, content-aware bit allocation [17, 25], *e.g.*, with a region of interest (ROI) [3, 23, 30] or importance map [25]. Second, rate adaptation is performed to adjust the bitrate during inference with a single trained model and move the operating point along the rate-distortion (RD) curve [8, 9, 21, 30, 36]. The rate adaptation approach used in this paper follows the latter option, with closest prior art [21], where an importance map with a feature map masking mechanism and a learned mask for content-aware bit allocation are used. However, our rate adaptation goes beyond [21], as it uses a masking mechanism to flexibly control the bitrate during inference with a single trained model.

3. Method

In this section, we will first introduce our proposed one-hot max (Ω) quantizer, then explain the bitrate adaptation mechanism, and finally show how it is embedded into the image compression system.

3.1. One-Hot Max (Ω) Quantization

Figure 1 shows how the residual $\boldsymbol{r}$ is reorganized by convolution with a 1×1 kernel, yielding C' feature maps. A subsequent reshaping splits the last dimension (size C') into a $C_{\mathcal{Z}}$-sized feature map dimension and a S-sized vectorial Ω quantizer input dimension, fulfilling $C' = C_{\mathcal{Z}} \cdot S$. Next, the representation is rectified and multiplied by $-\beta < 0$, where β is a learnable parameter to control the hardness of the backward approximation. Thereby, one obtains $\boldsymbol{z} = (\boldsymbol{z}_i) \in \mathbb{R}^{H_{\mathcal{R}} \times W_{\mathcal{R}} \times C_{\mathcal{Z}} \times S}$, with $\boldsymbol{z}_i = (z_{i,s}) \in \mathbb{R}^S$ being the feature map pixels with index $i \in \mathcal{I} = \{1, ..., H_{\mathcal{R}} \cdot W_{\mathcal{R}} \cdot C_{\mathcal{Z}}\}$. We observed that enforcing $z_{i,s} \leq 0$, following [1], yields slight performance advantages. Quantization is performed by our proposed vectorial one-hot max (Ω) quantization function

$$\hat{\boldsymbol{z}}_i = \boldsymbol{\Omega}(\boldsymbol{z}_i), \tag{1}$$

with $\hat{\boldsymbol{z}}_i = (\hat{z}_{i,s})$ and

$$\hat{z}_{i,s} = \begin{cases} 1 & \text{for } s = \text{argmax}(\boldsymbol{z}_i), \\ 0 & \text{else}, \end{cases} \tag{2}$$

taking the argmax over the elements $z_{i,s}$ of the input vector $\boldsymbol{z}_i$, with quantizer dimension index $s \in \mathcal{S} = \{1, ..., S\}$. To yield a differentiable function for the backward pass, we use the approximation

$$\tilde{\boldsymbol{z}}_i = \textbf{softmax}(\boldsymbol{z}_i), \quad \text{with } \tilde{\boldsymbol{z}}_i = (\tilde{z}_{i,s}), \tag{3}$$

thereby following [1, 14]. Finally, the quantized residual is reshaped back into its original size and again convolved to yield the quantizer output $\hat{\boldsymbol{r}}$.

Note that using convolutional layers to create and collapse the quantizer dimension S before and after the quantizer cannot be transformed into a lookup table operation as with a codebook. With a classical codebook, first, the distance between activation and codebook entries would be computed and second, the nearest codebook entry *from the same codebook* replaces the activation [1]. In our case, the two layers are decoupled and different in general. The Ω quantizer actually can be interpreted as a vector quantizer with one-hot codebook vectors in a higher-dimensional space (no need to store as codebook).

3.2. Rate Adaptation

To adapt the bitrate during inference, the number of feature maps in the bottleneck is dynamically varied in the range $[C_{\text{min}}, C_{\mathcal{Z}}]$ by masking the quantized representation $\hat{\boldsymbol{z}}$. Here, we consider the one-hot quantized data as $\hat{\boldsymbol{z}} = (\hat{\boldsymbol{z}}(c)) \in \{0,1\}^{H_{\mathcal{R}} \times W_{\mathcal{R}} \times C_{\mathcal{Z}} \times S}$, $\hat{\boldsymbol{z}}(c) \in \{0,1\}^{H_{\mathcal{R}} \times W_{\mathcal{R}} \times S}$ with feature map (FM) index $c \in \{1, ..., C_{\mathcal{Z}}\}$. The masked quantizer output is

$$\hat{\boldsymbol{z}}' = (\hat{\boldsymbol{z}}'(c)) = \mathbf{M}(\hat{\boldsymbol{z}}), \tag{4}$$

where the masking function $\mathbf{M}()$ is obtained by

$$\hat{\boldsymbol{z}}'(c) = \begin{cases} \hat{\boldsymbol{z}}(c) & \text{for } c \leq \lceil C_{\text{min}} + q \cdot (C_{\mathcal{Z}} - C_{\text{min}}) \rceil, \\ \mathbf{0} & \text{else}, \end{cases} \tag{5}$$

employing the quality hyperparameter $q \in [0, 1]$.

During training, the quality hyperparameter q is sampled from a probability density function $p_Q()$. Thus, we indirectly train with adaptive bitrate conditions where q can take on arbitrary values. Due to the nature of this sampling process, feature maps with low indices are trained more frequently than those with high indices (high-indexed feature

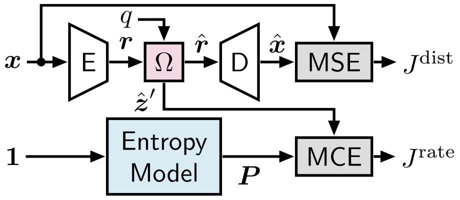


Figure 2. **Training configuration for learned image compression with one-hot max (Ω) quantization:** Autoencoder training with Ω quantizer (details in Figure 1) and entropy model by distortion and rate loss. The reference approach replaces the Ω quantizer by the standard rounding-based (INT) quantizer.

maps are only trained for high q values, while low-indexed feature maps are trained for both low and high q values). To combat this imbalance at least to some extent, we concentrate the probability density function $p_Q()$ on higher values of q, by sampling

$$q \sim p_Q(q) = \begin{cases} 2q & \text{for } 0 \leq q \leq 1, \\ 0 & \text{else,} \end{cases} \tag{6}$$

see also [31]. During inference, q is no longer sampled but freely chosen to control the rate-distortion trade-off.

3.3. Compression Architecture

Figure 2 shows the embedding of our proposed Ω quantizer into an autoencoder architecture with an entropy model estimating the bottleneck distribution (training). The integer-rounding-based (INT) reference quantizer is explained in Section A in the supplementary material. With the input image $\boldsymbol{x} \in [0,1]^{H_\mathcal{X} \times W_\mathcal{X} \times C_\mathcal{X}}$ and output image $\hat{\boldsymbol{x}} \in [0,1]^{H_\mathcal{X} \times W_\mathcal{X} \times C_\mathcal{X}}$, both of height $H_\mathcal{X}$, width $W_\mathcal{X}$, and $C_\mathcal{X} = 3$ color channels, the distortion loss

$$J^{\text{dist}} = \text{MSE}(\boldsymbol{x}, \hat{\boldsymbol{x}}) = \frac{\|\boldsymbol{x} - \hat{\boldsymbol{x}}\|_2^2}{H_\mathcal{X} \cdot W_\mathcal{X} \cdot C_\mathcal{X}} \tag{7}$$

minimizes the mean squared error (MSE) during training.

The entropy model predicts the prior probability for the occurrence of each of the S possible one-hot vectors in each feature map pixel of $\hat{\boldsymbol{z}}$. We use an entropy model based on convolutional layers such that a constant tensor $\boldsymbol{1} \in \{1\}^{H_\mathcal{R} \times W_\mathcal{R} \times 1}$ filled with ones, having a height and width dimension equal to the residual, generates distributions for arbitrary image sizes. A softmax activation restricts the output $\boldsymbol{P} = (P_{i,s}) \in [0,1]^{H_\mathcal{R} \times W_\mathcal{R} \times C_\mathcal{Z} \times S}$ to the interval $P_{i,s} \in [0,1]$ with $\sum_{s \in \mathcal{S}} P_{i,s} = 1$, see Section B in the supplementary material for details.

The entropy model minimizes the bitrate, simultaneously being the rate loss

$$J^{\text{rate}} = \text{MCE}(\hat{\boldsymbol{z}}', \boldsymbol{P}) = -\sum_{i \in \mathcal{I}} \sum_{s \in \mathcal{S}} \frac{\hat{z}'_{i,s} \cdot \log_2(P_{i,s})}{H_\mathcal{X} \cdot W_\mathcal{X}}, \tag{8}$$

realized as the mean cross entropy (MCE) between the masked quantized residual pixels $\hat{\boldsymbol{z}}'_i \in \{0,1\}^S$ and the estimated probability distribution $\boldsymbol{P}$. Accordingly, the accumulation over s in (8) is done effectively only over the non-masked feature maps. Hence, in combination with (7), we obtain the rate-distortion trade-off being controlled by the hyperparameter $\lambda \in]0,1[$ in the rate-distortion (RD) loss

$$J = \lambda \cdot J^{\text{dist}} + (1 - \lambda) \cdot J^{\text{rate}}, \tag{9}$$

whereby the RD trade-off is determined by

$$\lambda = \lambda_{\text{min}} + q \cdot (\lambda_{\text{max}} - \lambda_{\text{min}}), \tag{10}$$

ranging between a minimum and maximum rate-distortion weight λ_{min} and λ_{max}, respectively, depending on q.

The model is trained by the rate-distortion loss (9) with masked quantizer output $\hat{\boldsymbol{z}}'$ (4) as input to the decoder and adapted λ (10), where q is sampled randomly from $p_Q()$ (6).

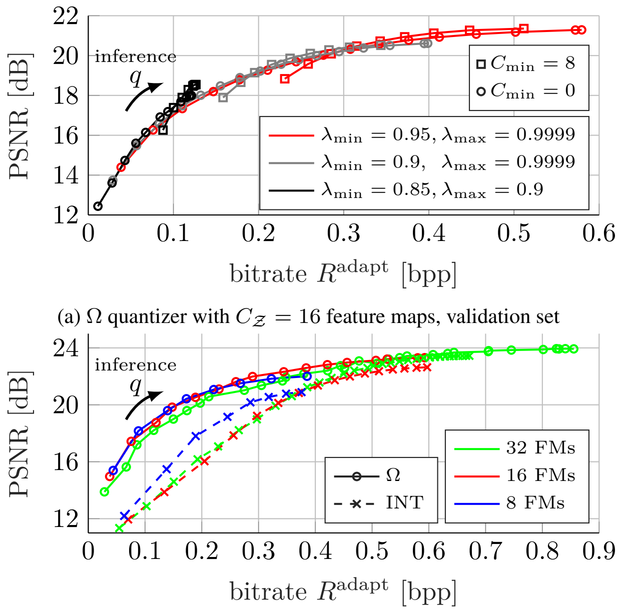
4. Evaluation

4.1. Experimental Setup

To prove transferability, we conduct experiments on MNIST [20] and Kodak [10], while training on MNIST (separate training split) and OpenImages [19], respectively. Details on the datasets and training hyperparameters can be found in Section C in the supplementary material. As our goal is to evaluate our novel Ω quantizer rather than using the latest state-of-the-art autoencoder, we defined network architectures allowing easy comparison described in Section D in the supplementary material. Using advanced and more complex network architectures, *e.g.*, a hyperprior architecture [6], the absolute compression performance would be expected higher for all investigated methods. We report on the peak signal-to-noise ratio (PSNR).

4.2. Experiments on MNIST

Figure 3(a) shows RD curves for the Ω quantizer on the MNIST validation set with $C_\mathcal{Z} = 16$ FMs and various configurations for $C_{\text{min}} \in \{0, 8\}$ and intervals of $\lambda \in \{[0.95, 0.9999], [0.9, 0.9999], [0.85, 0.9]\}$. Each setting results in a smooth RD curve, which we trace to a well structured bottleneck representation, being ordered by importance. Due to the lightweight network architectures we used, PSNR values range up to 22 dB (24 dB on test set). For $\lambda \in [0.95, 0.9999]$ and $C_{\text{min}} = 0$, good PSNR values at low and high bitrates can be obtained, while choosing $C_{\text{min}} = 8$ achieves slightly better trade-offs only in the higher bitrate regime. For $\lambda \in [0.85, 0.9]$, lower bitrates and best RD trade-offs in the low-bitrate regime are observed. However, bitrate adaptation schemes should aim at a wide range of good RD trade-offs. The lower C_{min} or the larger the λ interval, the larger the resulting bitrate range. The average value of λ further influences the trade-off to focus on specific bitrate regimes. In the following, we choose



(a) Ω quantizer with $C_{\mathcal{Z}} = 16$ feature maps, validation set

(b) Ω and INT quantizer with $C_{\mathcal{Z}}, C_{\mathcal{R}} \in \{8, 16, 32\}$ FMs, test set

Figure 3. **MNIST RD curves**. The average bitrate depends on the number of channels controlled by q during inference. (a) Ω quantizer with $S = 256$ and various choices of C_{min} and λ on the validation set. (b) Ω and INT quantizers with $C_{\text{min}} = 0$, $\lambda \in [0.95, 0.9999]$, $S = 256$ (for Ω quantization) on the test set.

$C_{\text{min}} = 0$ with $\lambda \in [0.95, 0.9999]$ to cover a wide bitrate range.

Figure 3(b) compares Ω and integer-rounding-based (INT) quantization for $C_{\mathcal{Z}}, C_{\mathcal{R}} \in \{8, 16, 32\}$ feature maps, respectively, on the MNIST test set. *The proposed Ω quantizer clearly outperforms the INT quantizer in the entire low bitrate regime (< 0.4 bpp) — and still performs a bit better at high bitrates — for each feature map configuration.* Interestingly, $C_{\mathcal{Z}}$ has no strong influence on the RD curve for the Ω quantizer as could have been expected. The largely overlapping curves show the robust well-behaving properties of the Ω quantizer. In contrast, the INT quantizer behaves less predictable and stable which can be seen at the RD curve for 8 FMs exceeding the RD curves of 16 and 32 FMs.

4.3. Experiments on Kodak

For the evaluation on Kodak, shown in Figure 4, we use $S = 32$ (for Ω quantization), $C_{\text{min}} = 0$, and an RD trade-off parameter $\lambda \in [0.95, 0.9999]$ during training. Again, PSNR values range up to 31 dB as we use simple architectures. *Ω quantization again excels INT quantization in PSNR at low bitrates*, while INT quantization is better at high bitrates above about 0.4 bpp. With decreasing bitrate, the INT quantizer significantly drops at 0.18 bpp from about 25 dB to below 22.5 dB in PSNR, while the Ω quantizer in the same bitrate range stays at a PSNR > 24 dB until reaching 0.1 bpp. We conclude that Ω quantization is better suited for the highly relevant low bitrate regime.

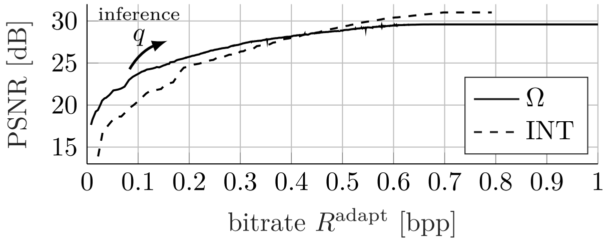


Figure 4. **Kodak RD curves**. Models are trained on OpenImages [19] and tested on Kodak [10]. The average bitrate depends on q as chosen during inference.

4.4. Final Discussion

Even though a model with rate adaptation of course does not reach the performance of multiple models each specifically trained for a dedicated RD trade-off, *e.g.*, [4, 6, 27], the rate adaptation mechanism still has essential advantages in training and inference, *e.g.*, faster training and inference and better comparability to compression standards which also frequently provide rate adaptation. Due to the masking being applied after encoding and quantization, the Ω quantizer can adjust to temporary bandwidth/bitrate shortages on the transmission channel *on the fly* — simply by removing feature maps at any point in the transmission network. Note that we do not compare our approach in absolute terms to prior works since we use less complex architectures, so lower PSNR is expected.

5. Conclusions

In this work, we proposed a learnable vector quantization approach called one-hot max (OHM, Ω) quantizer without codebook and showed how to use it in learned image compression with adaptive bitrate. We compare our Ω quantizer with integer-rounding-based (INT) quantization on MNIST and the conventional Kodak dataset. *Our proposed Ω quantizer excels the baseline at all bitrates on MNIST and at low bitrates (< 0.4 bpp) on Kodak.* Furthermore, it shows better generalizability and predictability for different bottleneck sizes and bitrates. Looking forward, by its rate adaptation mechanism, the Ω quantizer is perfectly suited for media transmission with flexible options to reduce the bitrate at any point during transmission.

Acknowledgement: The research leading to these results is funded by the German Federal Ministry for Economic Affairs and Energy within the project "KI Data Tooling – Methoden und Werkzeuge für das Generieren und Veredeln von Trainings-, Validierungs- und Absicherungsdaten für KI-Funktionen autonomer Fahrzeuge". The authors would like to thank the consortium for the successful cooperation.

References

[1] Eirikur Agustsson, Fabian Mentzer, Michael Tschannen, Lukas Cavigelli, Radu Timofte, Luca Benini, and Luc V. Gool. Soft-to-Hard Vector Quantization for End-to-End Learning Compressible Representations. In *Proc. of NeurIPS*, pages 1141–1151, Long Beach, CA, USA, Dec. 2017. 1, 2

[2] Eirikur Agustsson and Lucas Theis. Universally Quantized Neural Compression. In *Proc. of NeurIPS*, pages 12367–12376, Vancouver, BC, Canada, Dec. 2020. 2

[3] Hiroaki Akutsu and Takahiro Naruko. End-to-End Learned ROI Image Compression. In *Proc. of CVPR*, pages 4321–4325, Long Beach, CA, USA, June 2019. 2

[4] Johannes Ballé, Valero Laparra, and Eero P. Simoncelli. End-to-End Optimization of Nonlinear Transform Codes for Perceptual Quality. In *Picture Coding Symposium*, pages 1–5, Nuremberg, Germany, Dec. 2016. 4

[5] Johannes Ballé, Valero Laparra, and Eero P. Simoncelli. End-to-end Optimized Image Compression. In *Proc. of ICLR*, pages 1–27, Toulon, France, Apr. 2017. 1, 2, 7

[6] Johannes Ballé, David Minnen, Saurabh Singh, Sung Hwang, and Nick Johnston. Variational Image Compression With a Scale Hyperprior. In *Proc. of ICLR*, pages 1–47, Vancouver, BC, Canada, Apr. 2018. 2, 3, 4, 7, 8

[7] Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or Propagating Gradients Through Stochastic Neurons for Conditional Computation. *arXiv*, Aug. 2013. (1308.3432). 2

[8] Yoojin Choi, Mostafa El-Khamy, and Jungwon Lee. Variable Rate Deep Image Compression With a Conditional Autoencoder. In *Proc. of ICCV*, pages 3146–3154, Seoul, Korea, Oct. 2019. 2

[9] Ze Cui, Jing Wang, Bo Bai, Tiansheng Guo, and Yihui Feng. G-VAE: A Continuously Variable Rate Deep Image Compression Framework. In *arXiv*, pages 1–8, Apr. 2020. 2

[10] Eastman Kodak Company. Kodak Lossless True Color Image Suite. http://r0k.us/graphics/kodak/, 1999. [Online; accessed 2021-10-05]. 3, 4, 8

[11] Allen Gersho and Robert M. Gray. *Vector Quantization and Signal Compression*. Springer Science & Business Media, 2012. 2

[12] Robert Gray. Vector Quantization. *IEEE ASSP Magazine*, 1(2):4–29, Apr. 1984. 2

[13] Robert M. Gray. Quantization Noise Spectra. *IEEE Trans. on Information Theory*, 36(6):1220–1244, Nov. 1990. 2

[14] Zongyu Guo, Zhizheng Zhang, Runsen Feng, and Zhibo Chen. Soft then Hard: Rethinking the Quantization in Neural Image Compression. In *Proc. of the International Conference on Machine Learning*, pages 3920–3929, virtual, July 2021. 1, 2

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *Proc. of CVPR*, pages 770–778, Las Vegas, NV, USA, June 2016. 9

[16] Yueyu Hu, Wenhan Yang, Zhan Ma, and Jiaying Liu. Learning End-to-End Lossy Image Compression: A Benchmark. *Transactions on Pattern Analysis and Machine Intelligence (Early Access)*, pages 1–18, Mar. 2021. 1

[17] 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 *Proc. of CVPR*, pages 4385–4393, Salt Lake City, UT, USA, June 2018. 2

[18] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In *Proc. of ICLR*, pages 1–15, San Diego, CA, USA, May 2015. 8

[19] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. The Open Images Dataset V4: Unified Image Classification, Object Detection, and Visual Relationship Detection at Scale. *IJCV*, 128:1956–1956, Mar. 2020. 3, 4, 8

[20] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-Based Learning Applied to Document Recognition. *Proceedings of the IEEE*, 86(11):2278–2324, Nov. 1998. 3, 8

[21] Mu Li, Wangmeng Zuo, Shuhang Gu, Debin Zhao, and David Zhang. Learning Convolutional Networks for Content-Weighted Image Compression. In *Proc. of CVPR*, pages 3214–3223, Salt Lake City, UT, USA, June 2018. 2

[22] Yoseph Linde, Andrés Buzo, and Robert Gray. An Algorithm for Vector Quantizer Design. *IEEE Trans. on Communications*, 28(1):84–95, Jan. 1980. 2

[23] Jonas Löhdefink, Andreas Bär, Nico M. Schmidt, Fabian Hüger, Peter Schlicht, and Tim Fingscheidt. Focussing Learned Image Compression to Semantic Classes for V2X Applications. In *Proc. of IV*, pages 1370–1377, Las Vegas, NV, USA, Oct. 2020. 2

[24] Jonas Löhdefink, Fabian Hüger, Peter Schlicht, and Tim Fingscheidt. Scalar and Vector Quantization for Learned Image Compression: A Study on the Effects of MSE and GAN Loss in Various Spaces. In *Proc. of ITSC*, pages 1–8, Rhodes, Greece, Sept. 2020. 2

[25] Fabian Mentzer, Eirikur Agustsson, Michael Tschannen, Radu Timofte, and Luc Van Gool. Conditional Probability Models for Deep Image Compression. In *Proc. of CVPR*, pages 4394–4402, Salt Lake City, UT, USA, June 2018. 2

[26] Fabian Mentzer, Eirikur Agustsson, Michael Tschannen, Radu Timofte, and Luc Van Gool. Practical Full Resolution Learned Lossless Image Compression. In *Proc. of CVPR*, pages 10629–10638, Long Beach, CA, USA, June 2019. 8

[27] Fabian Mentzer, George D. Toderici, Michael Tschannen, and Eirikur Agustsson. High-Fidelity Generative Image Compression. In *Proc. of NeurIPS*, volume 33, pages 11913–11924, Vancouver, BC, Canada, Dec. 2020. 1, 2, 4, 7, 8

[28] Antonio Ortega and Kannan Ramchandran. Rate-Distortion Methods for Image and Video Compression. *Signal Processing Magazine*, 15(6):23–50, Nov. 1998. 1

[29] David Salomon. *Data Compression: The Complete Reference*. Springer Science & Business Media, 2004. 2

[30] Myungseo Song, Jinyoung Choi, and Bohyung Han. Variable-Rate Deep Image Compression Through Spatially-Adaptive Feature Transform. In *Proc. of ICCV*, pages 2380–2389, virtual, Oct. 2021. 2

[31] Masashi Sugiyama. *Introduction to Statistical Machine Learning*. Morgan Kaufmann, 2016. 3

[32] Lucas Theis, Wenzhe Shi, Andrew Cunningham, and Ferenc Huszár. Lossy Image Compression With Compressive Autoencoders. In *Proc. of ICLR*, pages 1–19, Toulon, France, Apr. 2017. 2

[33] Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural Discrete Representation Learning. In *Proc. of NIPS*, pages 6306–6315, Long Beach, CA, USA, Dec. 2017. 2

[34] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-Resolution Image Synthesis and Semantic Manipulation With Conditional GANs. In *Proc. of CVPR*, pages 8798–8807, Salt Lake City, UT, USA, June 2018. 9

[35] Will Williams, Sam Ringer, John Hughes, Tom Ash, David MacLeod, and Jamie Dougherty. Hierarchical Quantized Autoencoders. In *Proc. of NeurIPS*, pages 4524–4535, Vancouver, BC, Canada, Dec. 2020. 2

[36] Fei Yang, Luis Herranz, Joost van de Weijer, José A. Iglesias Guitián, Antonio M. López, and Mikhail G. Mozerov. Variable Rate Deep Image Compression with Modulated Autoencoder. *IEEE Signal Processing Letters*, 27:331–335, Jan. 2020. 2

[37] Yibo Yang, Robert Bamler, and Stephan Mandt. Improving Inference for Neural Image Compression. In *Proc. of NeurIPS*, pages 573–584, Vancouver, BC, Canada, Dec. 2020. 2

[38] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks. In *Proc. of ICCV*, pages 2223–2232, Venice, Italy, Oct. 2017. 9