Compact Latent Representation for Image Compression (CLRIC)

Ayman A. Ameen* †, Thomas Richter*, and André Kaup‡,

*Fraunhofer Institute for Integrated Circuits IIS, Erlangen, Germany
† Department of Physics, Faculty of Science, Sohag University, Egypt
‡ Friedrich-Alexander University at Erlangen-Nürnberg, Erlangen, Germany

Abstract—Current image compression models often require separate models for each quality level, making them resource-intensive in terms of both training and storage. To address these limitations, we propose an innovative approach that utilizes latent variables from pre-existing trained models (such as the Stable Diffusion Variational Autoencoder) for perceptual image compression. Our method eliminates the need for distinct models dedicated to different quality levels. We employ overfitted learnable functions to compress the latent representation from the target model at any desired quality level. These overfitted functions operate in the latent space, ensuring low computational complexity, around 25.5 MAC/pixel for a forward pass on images with dimensions (1363×2048) pixels. This approach efficiently utilizes resources during both training and decoding. Our method achieves comparable perceptual quality to state-of-the-art learned image compression models while being both model-agnostic and resolution-agnostic. This opens up new possibilities for the development of innovative image compression methods.

I. INTRODUCTION

Effective compression techniques for images and videos plays a pivotal role for supplying the high quality digital content without the need to update the current infrastructure. Currently, the entertainment services occupy a large portion of the Internet traffic. Such services require high-quality perceptual images and videos and often does not require accurate pixel-wise reconstruction. The current widely used techniques, such as HEVC [1] and VVC [2], relies on linear operators which are handcrafted to minimize the bitrate of the target. One major issue with these techniques is the production of unacceptable artifacts in the compressed image or video at low bit rate. Learned image compression promises to solve the issues of traditional compression techniques. Currently, there are numerous approaches for learned image compression, such as, generalizable autoencoder and overfitted learnable neural function.

Recent advancements in learned image compression have demonstrated impressive results in achieving low bit rates [3], [4] and high human visual quality [5], [6], utilizing either generalizable autoencoders or overfitted learnable functions. However, these techniques have some drawbacks and limitations. The generalizable autoencoder approach requires training entirely separate models for image compression, each tailored to specific quality levels with corresponding bit rates, which typically involves training 6-8 distinct models from scratch, each representing a different quality level.

Similarly, the overfitted learnable function method [7], [8]

attempts to fit a single image into a learnable function without leveraging prior information. This approach requires retraining from scratch for every new image and is performed in the image space, which is significantly larger than the latent space.

Our objective is to address these limitations through a novel approach. We trained an overfitted learnable function (such as COOLCHIC) on latents generated by existing latent image models (e.g., a stable diffusion autoencoder), without requiring retraining or backpropagation within the large pretrained latent image model. Our approch yields results comparable to state-of-the-art bit rate and image quality benchmarks, as shown in Figure 1. The advantages of this novel approach are summarized as follows:

- **Utilization of Pre-existing Models:** Leverage existing pretrained image generation models like the stable diffusion autoencoder for image compression without retraining or backpropagation, thus avoiding resource-intensive training processes.
- **Efficient Latent Variable Storage:** Efficiently store latent variables from the image generation model, achieving target quality and corresponding bit rates without needing to train a hyper-encoder and decoder.
- **Continuous Quality Adaptation:** Achieve continuous quality representation with target bit rates adapted to specific applications through adaptation of the loss function parameters, eliminating the need for separate models for each level of image quality.
- **Perceptual Quality Optimization:** Ensure perceptual quality is adaptive to the human visual system, enhancing the viewing experience, as depicted in Figure 5
- **Resource-efficient Overfitting:** Overfit the learnable function in the latent space instead of the image space, reducing memory and computational resource requirements, since latent space is substantially smaller than image space by a factor of eight or more, enabling tailored image compression for each model using stochastic gradient descent techniques.

II. RELATED WORKS

A. Generalizable Autoencoder

Since traditional image compression algorithms such as JPEG 2000 usually utilize only linear decorrelation transformations of the input data, a new neural image compression approach was proposed using autoencoder which is trained on large

images or videos dataset. [3], [4], [9], [10]. This approach utilizes the ability of the autoencoder to learn the nonlinear mapping of the images to high-dimensional low entropy latent space and then project them back to the image space [11], [12].

The autoencoder framework is based on four main components [3], [4]. The first one consisted of analysis neural transfer function g_a to map the image to latent space, followed by a quantization function Q, to quantize the latent. To restore the image back, a synthesis transform g_s is used. One crucial component is a Context-based Entropy model to reduce and estimate the bit-rate.

The network architecture of the analysis and synthesis transform functions may consist only of convolutional layers [13]. Another architecture introduced residual connection with convolutional layer as base model [14]. Also, transformer-based layers [15], and attention layers mixed with convolutional layers [16] have been utilised as transform functions.

Since the conditional entropy of the quantized latent vector $\hat{y}$ given a context is smaller than or equal to the entropy of $\hat{y}$ without context, several methods have been proposed for contextual entropy model. A hyper-prior analysis model h_a was proposed to extract a context from $\hat{y}$ as z, which is quantized to $\hat{z}$. a hyper-prior synthesis model h_s creates from $\hat{z}$ parameters of univariate Gaussians which model the probability distribution of $\hat{y}$ for the sake of entropy coding [12]. Other methods have extended the distribution model with Gaussian with mean and scale [17]. Other methods introduced asymmetric Gaussian distribution, Gaussian mixture distribution model [16], and a Gaussian-Laplacian logistic mixture model [18] as a distribution model that allows more flexibility.

To improve the compression efficiency, various context-based entropy modeling was proposed. An autoregressive model was introduced that conditioned each pixel with the previously decoded pixels for more effective context modeling [17]. Another context model is the checkerboard convolution which divides the Latent representation into anchor part which is used to extract the context for non-anchor part [16]. A channel-wise context model [19] and channel-wise with unevenly grouped context model [4] were introduced to exploit redundancy between channels. Recently, a context model which tried to exploit diverse range of correlations within the latent representation using attention-based architecture [3], [9].

B. Overfitted Neural Function

The notion of utilizing an overfitted neural function centers around representing image data as the learnable parameters of a neural function, as opposed to discrete pixel values. This neural function can be evaluated to reconstruct the RGB values of image pixels. The concept gained prominence with the advent of neural radiance fields, a technique for neural rendering [20]–[22]. In the domain of imaging, attempts have been made to represent entire datasets, such as the MNIST dataset, using neural functions to achieve resolution-agnostic representations [23]. One key advantage of modeling images as neural functions is their resolution agnosticism; images

can be represented as continuous neural functions and then evaluated at any desired resolution. This approach assumes that the image signals are inherently continuous.

The pioneering work that introduced the idea of overfitted learnable functions for image compression was COIN [8]. COIN employs a simple multilayer perceptron (MLP) to map pixel coordinates to their corresponding RGB values by leveraging the efficiency of periodic activation functions [24]. Although COIN achieved performance comparable to JPEG compression, it was limited by its inability to exploit pixel locality due to the non-local nature of MLPs. This limitation was mitigated by employing a multi-resolution latent representation followed by a non-linear MLP [21].

Drawing inspiration from multi-resolution latents, COOL-CHIC introduced an overfitted learned image codec with low decoding complexity that significantly enhanced compression performance compared to COIN.

The COOL-CHIC framework [7], [25]–[27] consists of four principal components: a multi-resolution latent representation followed by an upsampling kernel, either learned or predefined, subsequently followed by synthesis convolution layers incorporating residual connections. The final component is an autoregressive model that generates a probability distribution for each latent pixel based on previously decoded ones.

III. METHOD

A. Overall Architecture

The objective of our work is to introduce an architecture that has the capability representing latent variables from different image models in a compact and efficient manner. Our innovative approach leverages compact latent representations from pre-trained models to generate outputs with various bit rates. An overview of our proposed architecture is illustrated in Figure 1. This architecture allows for mapping any latent variable from a learned image model to a specified bit rate. In this configuration (Figure 1), an input image x is processed by a pretrained encoder E_θ with parameters θ, resulting in a latent variable y. The latent variable serves as desired target output of our learnable function f_ψ with parameters ψ, which is overfitted to approximate the latent under constraints regarding the target bit rate and its corresponding quality. The loss function is defined as follows [26], [27]:

$$\mathcal{L}(\psi) = D(y, \hat{y}) + \lambda R(\hat{\psi}) \tag{1}$$

where $R(\hat{\psi})$ represents the rate of the learnable parameters of the overfitted function used to depict the latent $\hat{y}$, λ is a Lagrange multiplier that can be tuned to control quality for achieving specific or targeted bit rates, and D is the mean square error in latent Space. Following the training phase, the latent variable $\hat{y}$ is represented by the weights of a learnable function f_ψ, where ψ denotes the parameters of this function. These parameters are stored or transmitted to the decoder. The function f_ψ is evaluated to obtain the latent representation $\hat{y}$ at the decoder side. This latent representation is feed into a decoder D with pretrained learnable parameters ϕ, resulting in the reconstructed image $\hat{x}$.
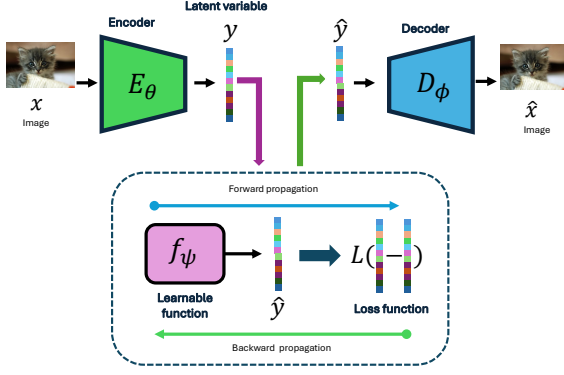
Fig. 1: General overview proposed approach of overfitting a learnable function on the latent from generalizable autoencoder

B. Generative Generalizable Autoencoder

The generative generalizable autoencoder (E_θ, D_ϕ) selected from the Latent Diffusion autoencoder family [28]–[30], which is pretrained for synthesizing high-resolution images. This selection is based on the models' capability for perceptual compression rather than semantic compression. This capability allows for the generation of images with high perceptual quality at a reduced bit rate. Another advantage of these models is their ability to capture the full data distribution by mapping images from pixel space to latent space with a reduced number of parameters [31]–[33]. This parameter reduction enables the overfitted learnable function to operate effectively with fewer parameters, thereby minimizing training resource requirements.
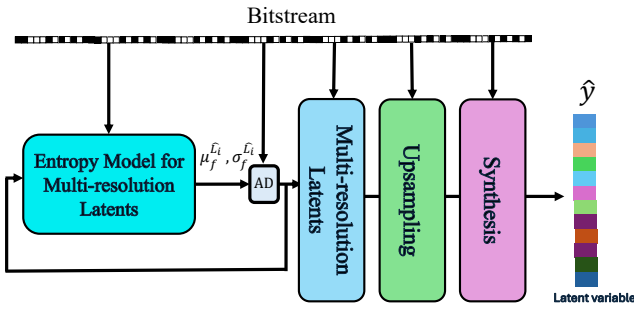
C. Overfitted Neural Function

Fig. 2: General overview of the proposed overfitted learnable decoder architecture

We utilized COOLCHIC architecture [7], [26], [27] as our overfitted learnable function f_ψ as illustrated in Figure 2. The proposed architecture consists of three main neural networks [7], [26]. The first neural network is an autoregressive model for multi-resolution latents, denoted as f_L, which provides a probability model characterized by a mean $\mu_f^{\hat{L}_i}$ and a standard deviation $\sigma_f^{\hat{L}_i}$, given the previously decoded latents. The

primary objective of the autoregressive process is to enhance the contextual information available for entropy coding. This is because the conditional entropy of $\hat{y}$ given a specific context is less than or equal to the overall entropy of $\hat{y}$:

$$H(\hat{y}) \leq H(\hat{y}|context)$$

The context can be leveraged to achieve better compression by reducing the bit-rate needed to encode the data. The multi-resolution latents $\hat{L}$ consist of two-dimensional grids with decreasing size according to:

$$\hat{L} = \left\{ \hat{L}_k \in \mathbb{Z}^{\frac{H}{2^k} \times \frac{W}{2^k}}, k \in 0, \dots, K-1 \right\}.$$

Here, H and W represent the width and height of a feature map of the latent from the generalization model, and K is the number of the multi-resolution grids. The second component is an adaptive up-sampling network f_u, which is initialized with a bicubic sampling kernel and updated during training to adapt to the targeted data [26]. The final network is a synthesis neural network f_ψ, which produces the final the latent $\hat{y}$ of the generalizable autoencoder.

D. Effective Bit-Rate Aware Training of Overfitted Neural Function

The objective of the optimization process is to minimize the distortion between the latent variable y and its reconstructed latent $\hat{y}$, while also reducing the total bit rate associated with all the learnable parameters (latent and model) ψ of the overfit learnable function f_ψ. This optimization aims to minimize the overall loss, which is expressed in 1 [3], [7], [26]. The optimization process occurs for every single source image to be encoded as a separate entity. To reduce the bit rate, quantization plays a crucial role in reducing the bitstream entropy by introducing a lossy compression scheme. However, quantization is a non-differentiable process. To address this issue, we approximate the quantization by adding uniform noise and switch to the actual quantization at the end of the training process of the overfitted neural function f_ψ [3], [7], [9].

IV. EXPERIMENTS

Initially, we outline our configuration and setup. Subsequently, we evaluate the model's performance through visual results followed by quantitative analysis using perceptual quality metrics and classical distortion metrics. Finally, we demonstrate the capability of continuous quality representation of the latent space at any arbitrary bitrate. Finally, we present our framework's complexity and limitations.

A. Configuration

This subsection presents the configuration setup for our generalizable autoencoder and our overfitted learnable function. These configurations can be easily modified depending on the target generalizable autoencoder model.
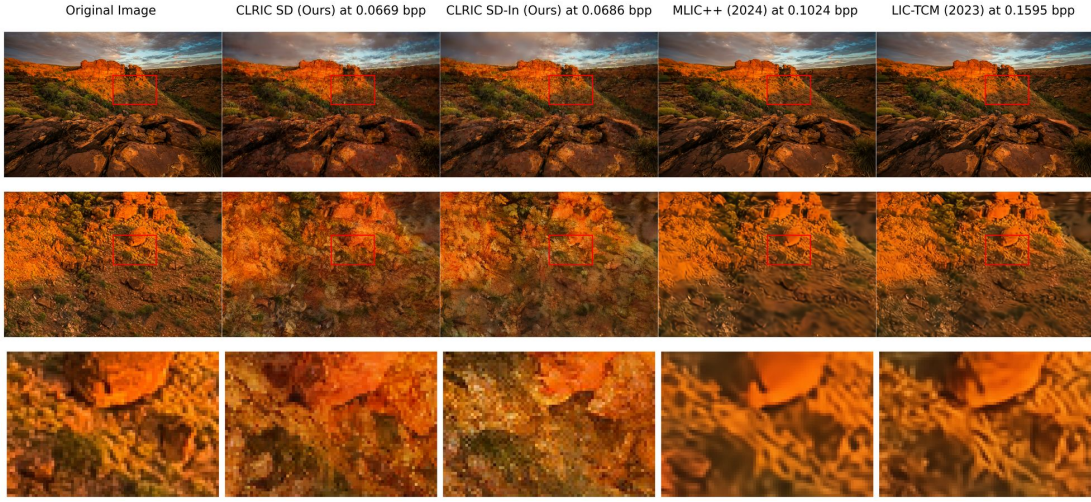
Fig. 3: Our novel approach performance compared to MLIC++ and LIC-TCM on image num. 27 from the CLIC Professional Valid 2020 dataset.

Fig. 4: Comparison between our approach and different models on image num. 7 from the Kodak dataset.

1) Generalizable Autoencoder: We select three derivatives from well-known and widely used open-source pretrained generative autoencoder models. The first model is Stable Diffusion v1.4 autoencoder, referred to in our results as SD [28]. The second and third models are Stable Diffusion XL refiner and Stable Diffusion 2 inpainting autoencoders [29], [30], which we refer to as SD-XL and SD-In, respectively.

2) Overfitted Learnable Function: The overfitted learnable function is based on the COOL-CHIC framework [7], [26], which offers a framework with low decoding complexity. In our experimental setup, we employ an auto-regressive model characterized by an 8-context window. The model outputs the probability distribution for predicting the next latent variable to be decoded. This auto-regressive model comprises three layers with dimensions 8, 8, and 2, respectively, and incorporates a residual connection between the first and second layers. The two-dimensional multi-resolution latent variables are structured into seven grids. The upsampling network utilizes transposed convolutions with a learnable kernel of size 8, initially configured as a bicubic upsampling kernel [26]. The synthesis network is composed of four layers with residual connections every two layers. The output corresponds to the latent size of a Generalizable Autoencoder, with the number of channels matching that of the Generalizable Autoencoder's latent channels.

3) Training Strategy: The training strategy for this overfitted function is grounded in the COOL-CHIC framework [7], [26], [27], starting with an initial learning rate of 10^{-2} and employing a decaying learning schedule. Training consists of 15,900 iterations divided into warm-up and main training phases. The warm-up phase includes 2,800 iterations split among five candidates trained over 400 iterations each; the top two candidates undergo an additional 400 iterations. Ultimately, the best candidate proceeds to the main training phase for an additional 13,100 iterations.

B. Performance

To evaluate the performance and generalization capability of our model, we conducted validation using two distinct datasets: the Kodak Dataset and CLIC Professional Valid 2020. The Kodak Dataset, widely used for validating image compression models, consists of 24 images. Meanwhile, CLIC Professional Valid 2020 comprises 41 high-resolution images,
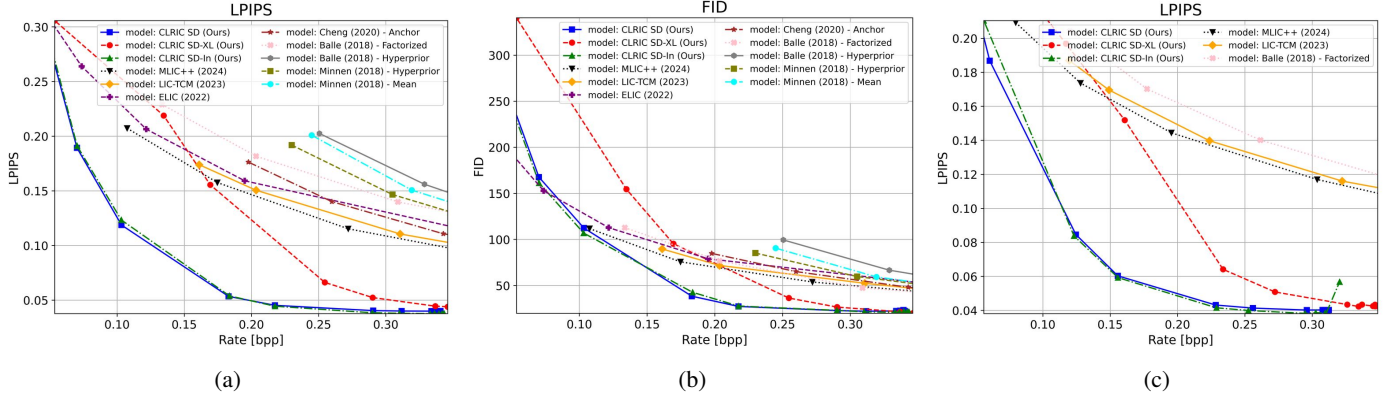
Fig. 5: Assessments and comparisons of image compression models using different metrics and datasets. (a) LPIPS scores on the Kodak dataset, (b) FID scores on the Kodak dataset, and (c) LPIPS scores on the CLIC Professional Valid 2020 dataset.

making it well-suited for evaluating models in the context of modern digital imagery. We benchmarked our model against these datasets and compared its performance with various other models. We compared our results to those of several state-of-the-art learned image compression models, including MLIC++ [3], LIC-TCM [14], and ELIC [4]. Additionally, we considered various model variants from notable works such as the Balle (2018) models, specifically the Factorized and Hyperprior variants [12]. We also included the Minnen (2018) model variants, namely Mean and Hyperprior [17], as well as Cheng (2020) variants, which encompass Anchor and Attention [16].

1) Visual Results: Our approach employs a Variational Autoencoder (VAE) trained with a perceptual compression strategy, resulting in higher image fidelity and a pleasant perceptual visual experience. This method maintains high granularity and accurately captures the prominent structures in natural images. In contrast, competing models at comparable bitrates often exhibit blurred details or smooth structures with missing granularity. At low bitrates, where other models tend to produce overly smooth outputs lacking fine details, our method maintains detailed granularity, as shown in Figure 3. Our technique effectively reproduces elements like wavy water surfaces and cloudy scenes more accurately compared to other methods that yield overly smooth results. Our method excels in preserving detail and color fidelity, as illustrated in Figure 4. The synthetic details generated by our model closely resemble the original image's color and texture, whereas other models often suffer from noticeable color shifts and lack of detail. At higher bitrates, our approach retains rich textures that other models' images frequently miss.

2) Perceptual Quantitative Analysis: Our method employs a perceptual image compression strategy, necessitating an analysis of its performance using perceptual metrics. We compared our model against state-of-the-art learned image compression models using the Kodak dataset and the CLIC Professional Validation 2020 dataset (Figure 5). The evaluation focused on perceptual quality, quantified by the Learned Perceptual Image Patch Similarity (LPIPS) metric, in relation to bit rate, denoted in bits per pixel (bpp). LPIPS was selected due to its strong correlation with human visual perception, aligning

well with our perceptual image compression approach. Perceptual quality was assessed for our approach across three different variational autoencoder models. We observed that our approach using the SD and SD-In variational autoencoders as front-end exhibited similar performance levels, with stability diminishing as bit rate increases, as illustrated in Figure 5. Meanwhile, the SD-XL based approach demonstrated slightly worse performance and required higher bit rates to achieve comparable image quality. Other models demanded significantly higher bit rates to match similar levels of perceptual quality. Furthermore, we utilized the Fréchet Inception Distance (FID) to evaluate image quality. FID measures the similarity between the distributions of reconstructed images and original images. Our method achieved superior results in terms of FID, as depicted in Figure 5. The SD-XL autoencoder-based approach required a higher bit rate to reach an equivalent FID compared to approaches using SD or SD-In.

V. Conclusion

In this paper, we presented a novel image compression method based on a two-stages approach. The proposed approach consists of two main components: a generalizable autoencoder capable of leveraging pre-trained generative image models, and an overfitted function that efficiently encodes the latent representation in a compact form. Our method is efficient resource utilization by operating in the latent space rather than the image space. Furthermore, our method allows for continuous quality representation at any target bitrate. We performed analyses using the Learned Perceptual Image Patch Similarity (LPIPS) and Fréchet Inception Distance (FID) metrics. Our approach achieved superior LPIPS and FID scores at comparable bit rates when compared to other models.

References

[1] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) Standard" vol. 22, pp. 1649–1668, Dec. 2012.

[2] B. Bross, J. Chen, J.-R. Ohm, G. J. Sullivan, and Y.-K. Wang, "Developments in International Video Coding Standardization After AVC, With an Overview of Versatile Video Coding (VVC)" vol. 109, pp. 1463–1493, Sep. 2021.

[3] W. Jiang, J. Yang, Y. Zhai, F. Gao, and R. Wang. "MLIC++: Linear Complexity Multi-Reference Entropy Modeling for Learned Image Compression." version 9. (Feb. 20, 2024).

[4] D. He, Z. Yang, W. Peng, R. Ma, H. Qin, and Y. Wang. "ELIC: Efficient Learned Image Compression with Unevenly Grouped Space-Channel Contextual Adaptive Coding." (Mar. 29, 2022).

[5] B. Chen, S. Yin, P. Chen, S. Wang, and Y. Ye. "Generative Visual Compression: A Review." (Feb. 3, 2024).

[6] H. Li and X. Zhang, "Human–Machine Collaborative Image Compression Method Based on Implicit Neural Representations" vol. 14, pp. 198–208, Jun. 2024.

[7] T. Ladune, P. Philippe, F. Henry, G. Clare, and T. Leguay, "COOL-CHIC: Coordinate-based Low Complexity Hierarchical Image Codec," presented at the Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 13 515–13 522.

[8] E. Dupont, A. Goliński, M. Alizadeh, Y. W. Teh, and A. Doucet. "COIN: COmpression with Implicit Neural representations." (Apr. 10, 2021).

[9] W. Jiang, J. Yang, Y. Zhai, P. Ning, F. Gao, and R. Wang. "MLIC: Multi-Reference Entropy Model for Learned Image Compression." version 9. (Jan. 16, 2024).

[10] L. Theis, W. Shi, A. Cunningham, and F. Huszár. "Lossy Image Compression with Compressive Autoencoders." (Mar. 1, 2017).

[11] J. Ballé, V. Laparra, and E. P. Simoncelli. "End-to-end Optimized Image Compression." (Mar. 3, 2017).

[12] J. Ballé, D. Minnen, S. Singh, S. J. Hwang, and N. Johnston. "Variational image compression with a scale hyperprior." (May 1, 2018).

[13] J. Liu, D. Liu, W. Yang, S. Xia, X. Zhang, and Y. Dai, "A Comprehensive Benchmark for Single Image Compression Artifact Reduction" vol. 29, pp. 7845–7860, 2020.

[14] J. Liu, H. Sun, and J. Katto, "Learned Image Compression With Mixed Transformer-CNN Architectures," presented at the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 14 388–14 397.

[15] M. Lu, P. Guo, H. Shi, C. Cao, and Z. Ma. "Transformer-based Image Compression." (Nov. 12, 2021).

[16] Z. Cheng, H. Sun, M. Takeuchi, and J. Katto. "Learned Image Compression with Discretized Gaussian Mixture Likelihoods and Attention Modules." (Mar. 30, 2020).

[17] D. Minnen, J. Ballé, and G. D. Toderici, "Joint Autoregressive and Hierarchical Priors for Learned Image Compression," in *Advances in Neural Information Processing Systems*, vol. 31, 2018.

[18] H. Fu, F. Liang, J. Lin, *et al.*, "Learned Image Compression With Gaussian-Laplacian-Logistic Mixture Model and Concatenated Residual Modules" vol. 32, pp. 2063–2076, 2023.

[19] J. Liu, G. Lu, Z. Hu, and D. Xu. "A Unified End-to-End Framework for Efficient Deep Image Compression." (May 23, 2020).

[20] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng. "NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis." (Aug. 3, 2020).

[21] T. Müller, A. Evans, C. Schied, and A. Keller, "Instant Neural Graphics Primitives with a Multiresolution Hash Encoding" vol. 41, pp. 1–15, Jul. 2022.

[22] B. Kerbl, G. Kopanas, T. Leimkuehler, and G. Drettakis, "3D Gaussian Splatting for Real-Time Radiance Field Rendering" vol. 42, pp. 1–14, Aug. 2023.

[23] H. Kim, A. Mnih, J. Schwarz, *et al.* "Attentive Neural Processes." (Jul. 9, 2019).

[24] V. Sitzmann, J. Martel, A. Bergman, D. Lindell, and G. Wetzstein, "Implicit Neural Representations with Periodic Activation Functions," in *Advances in Neural Information Processing Systems*, vol. 33, 2020, pp. 7462–7473.

[25] S. Lee, J.-B. Jeong, and E.-S. Ryu, "Entropy-Constrained Implicit Neural Representations for Deep Image Compression" vol. 30, pp. 663–667, 2023.

[26] T. Leguay, T. Ladune, P. Philippe, G. Clare, F. Henry, and O. Déforges, "Low-Complexity Overfitted Neural Image Codec," in *2023 IEEE 25th International Workshop on Multimedia Signal Processing (MMSP)*, Sep. 2023, pp. 1–6.

[27] T. Blard, T. Ladune, P. Philippe, G. Clare, X. Jiang, and O. Déforges. "Overfitted image coding at reduced complexity." (Mar. 18, 2024).

[28] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-Resolution Image Synthesis with Latent Diffusion Models," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, New Orleans, LA, USA, Jun. 2022, pp. 10 674–10 685, ISBN: 978-1-66546-946-3.

[29] D. Podell, Z. English, K. Lacey, *et al.* "SDXL: Improving Latent Diffusion Models for High-Resolution Image Synthesis." (Jul. 4, 2023).

[30] A. Sauer, F. Boesel, T. Dockhorn, A. Blattmann, P. Esser, and R. Rombach. "Fast High-Resolution Image Synthesis with Latent Adversarial Diffusion Distillation." (Mar. 18, 2024).

[31] B.-K. Kim, H.-K. Song, T. Castells, and S. Choi. "BK-SDM: A Lightweight, Fast, and Cheap Version of Stable Diffusion." (Nov. 16, 2023).

[32] S. H. Chan. "Tutorial on Diffusion Models for Imaging and Vision." (Sep. 6, 2024).

[33] G. M. Garcia, K. A. Zeid, C. Schmidt, D. de Geus, A. Hermans, and B. Leibe. "Fine-Tuning Image-Conditional Diffusion Models is Easier than You Think." (Sep. 17, 2024).