LoC-LIC: Low Complexity Learned Image Coding Using Hierarchical Feature Transforms

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 learned image compression models typically exhibit high complexity, which demands significant computational resources. To overcome these challenges, we propose an innovative approach that employs hierarchical feature extraction transforms to significantly reduce complexity while preserving bit rate reduction efficiency. Our novel architecture achieves this by using fewer channels for high spatial resolution inputs/feature maps. On the other hand, feature maps with a large number of channels have reduced spatial dimensions, thereby cutting down on computational load without sacrificing performance. This strategy effectively reduces the forward pass complexity from 1256 kMAC/Pixel to just 270 kMAC/Pixel. As a result, the reduced complexity model can open the way for learned image compression models to operate efficiently across various devices and pave the way for the development of new architectures in image compression technology.**

I. Introduction

Recently, Learned image compression models have achieved significant gains in bit rate reduction; however, traditional image and video compression methods, such as HEVC [1] and VVC [2], still largely dominate the field. Despite the potential benefits of the learned compression methods, the adaptation of these methods has been slow. One main reason is their high complexity and considerable resource requirements.

Most learned image compression architectures use transform analysis and synthesis with convolutional layers, maintaining a fixed number of channels and resizing the input for each (residual) layer. However, the total multiply-accumulate operations (MACs) per pixel increase with larger input sizes. Using a fixed number of channels results in higher MACs in initial layers without providing substantial benefits. To address this, we propose a hierarchical feature extraction approach with lower complexity by employing fewer channels for larger feature maps and more channels for smaller feature maps.

Our novel approach utilizes hierarchical feature extraction transforms to map images from the pixel domain to the latent domain and vice versa, reducing both memory and computational complexity. The key features of our approach include:

- Low complexity autoencoder through our novel hierarchical feature extraction, which has progressively deeper feature representations with a lower number of feature maps for larger sizes and higher features for smaller sizes, allowing reduction forward pass complexity from 1256 kMAC/Pixel to only 270 kMAC/Pixel.
- Hyper-autoencoder with multi-reference entropy model maintaining competitive performance to the state-of-the-art models.
- A large dataset that spans the large part of the image space manifold.

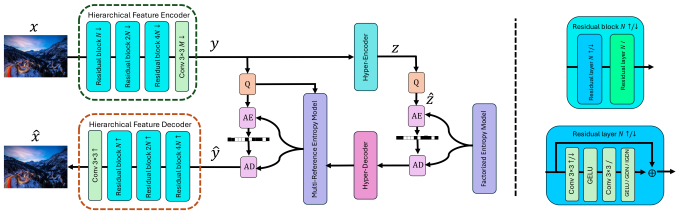


Fig. 1: Overall view of the proposed architecture with hierarchical feature encoder and decoder.

II. Related Works

Traditional image compression algorithms, such as JPEG 2000, often lack flexibility of non-linear mapping to input data. To address these limitations, a novel learned image compression method using autoencoders has been developed. This approach involves training an autoencoder on large datasets of images or videos [3]–[6].

The key advantage of this method lies in the autoencoder's capacity to map images into a low-entropy, high-dimensional latent space, and subsequently reconstruct them back into the image space [7], [8]. The network architecture is typically composed of analysis and synthesis transform functions, which can be implemented using pure convolutional layers with fixed number of channels N, [9]. Some architectures incorporate residual connections with convolutional layers as the base model [10], while others utilize transformer-based layers [11] or combine attention mechanisms with convolutional layers for enhanced performance [12]. The goal is for the reconstructed image $\hat{x}$ to closely resemble the original image x while minimizing the bit-rate R used for the latent representation. Generally, higher bit-rates result in lower distortion and vice versa. Therefore, optimization aims to balance distortion $D(x, \hat{x})$ with entropy measured in bitrate R. A Lagrangian multiplier is employed to manage this trade-off between distortion and target bit-rate [12], [13]. To reduce the bitrate, a quantization method is commonly such as a straight-through estimator (STE) is used. Another quantization approach is substituting deterministic rounding with stochastic

rounding, which has been shown to yield better results [8], [14]. Stochastic rounding can be easily implemented by adding uniform noise to the unquantized values. A hybrid approach that combines both straight-through estimation and stochastic rounding also exists [3]. Utilizing context information allows for more efficient data compression by reducing the bit-rate necessary for encoding. Context-based entropy models exploit surrounding or neighboring information to better predict and compress the current data. This strategy is particularly important in neural image compression, as it enables accurate bit-rate estimation while minimizing redundancy.

To enhance compression efficiency, various context-based entropy models have been proposed. An autoregressive model was introduced to condition each pixel on previously decoded pixels for more effective context modeling [14]. Another approach is the checkerboard convolution, which divides the latent representation into anchor and non-anchor parts, using the anchor part to extract context for the non-anchor part [12]. Furthermore, channel-wise context models [15], and channel-wise models with unevenly grouped contexts [5], have been developed to exploit redundancy between channels. Recently, an attention-based architecture has been proposed to capture a diverse range of correlations within the latent representation [3], [4].

Another promising approach for learned image compression involves using an overfitted neural network to represent image data as a continuous neural function instead of discrete pixel values. This neural function can be evaluated to reconstruct the RGB values of image pixels. Various efforts have been made to represent entire datasets, such as MNIST, using neural functions for resolution-agnostic representations [16]–[18]. A significant advantage of modeling images as neural functions is their resolution agnosticism: images are represented continuously and can be evaluated at any desired resolution. This approach assumes that image signals are inherently continuous.

The COIN framework [19] introduced the concept of using overfitted learnable functions for image compression. It utilizes a straightforward multilayer perceptron (MLP) to map pixel coordinates to their respective RGB values by effectively using periodic activation functions [20]. While COIN's performance was on par with JPEG compression, it was constrained by its inability to take advantage of pixel locality due to the inherently non-local characteristics of MLPs. This issue was addressed by employing a multi-resolution latent representation followed by a non-linear MLP [21]. COOL-CHIC [22]–[25] introduced an advanced overfitted learned image codec with reduced decoding complexity, which significantly improved compression efficiency compared to COIN.

III. Method

A. Motivation

Recent advancements in learned image compression models have significantly reduced bit rates. Despite these advancements, the integration of such models into existing systems and devices has been very slow. A primary reason for this slow adoption is the high computational complexity associated with these models, which demand substantial GPU memory and exhibit high operation complexity in terms of multiply-accumulate operations (MACs) per pixel.

In this paper, we introduce an innovative model designed to decrease both computational complexity and GPU memory usage by implementing hierarchical feature extraction from images. Hierarchical feature representation has been successfully applied across various domains, including generative image synthesis, super-resolution imaging, and various medical applications for segmentation and recognition [26]–[29]. However, many of these methods connect basic features from initial layers with more complex features from subsequent layers, leading to increased computational costs. Our approach overcomes this challenge by directly processing composite features while limiting the number of basic features and increasing the number of composite features to achieve computational efficiency.

B. Architecture Overview

The architecture we propose closely resembles mainstream learned image codecs [3], [12], [14], as depicted in Figure 1. The architecture consists of a Hierarchical Feature Encoder that functions as an analysis transform $g_{a_{hf}}$, organizing image data into a latent space hierarchically. This is followed by a quantization function Q, which quantizes the latent representation. It also incorporates a hyper-encoder and hyper-decoder featuring a multi-reference entropy model derived from MLIC++ [3], employing both local and global spatial contexts alongside channel and checkerboard attention context models to effectively capture correlations with linear complexity to minimize bit-rate usage.

The reconstruction of the image employs a synthesis transform $g_{s_{hf}}$. The process can be mathematically formulated as follows:

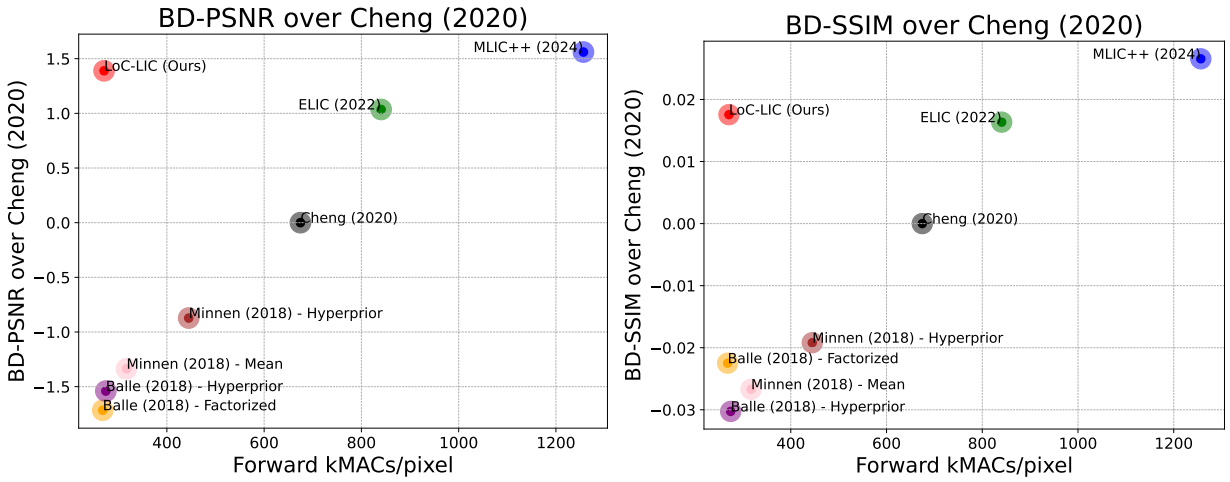
$$y = g_{a_{hf}}(x, \theta), \hat{y} = Q(y), \hat{x} = g_{s_{hf}}(\hat{y}, \phi)$$

where x represents the input image; y is the unquantized latent vector; $\hat{y}$ is the quantized latent vector; θ comprises parameters of the analysis transform function; and ϕ includes parameters for the synthesis transform function.

C. Hierarchical Feature Transform

Our hierarchical feature architecture efficiently transforms an input image of height H and width W into progressively deeper feature representations [30]. Initially, the input image is mapped to basic features with N channels, while height and width are reduced by a factor of 2. In the following layers, features map the number of channels to $2N$ is doubled while their dimensions are reduced by half, effectively enhancing the feature complexity while reducing spatial resolution. This process continues through both encoder and decoder layers.

The mathematical relationship governing this transformation across two sequential layers is expressed as:



(a) (b)

Fig. 2: The compression efficiency vs complexity of different learned image compression models. The complexity is measured in terms of (kMAC/Pixel). (a) BD-PSNR (b) BD-SSIM.

$$\text{out}_{i+1}\,(2C_i, H_i/2, W_i/2) = \text{in}_i(C_i, H_i, W_i)$$

where out_{i+1} represents the output of layer $i+1$, and $\text{in}_i$is the input from layer i, characterized by C_i channels with dimensions $H_i \times W_i$.

This architectural design significantly reduces computational complexity. High spatial resolution inputs/feature maps use fewer channels, thereby minimizing computational requirements. Meanwhile, feature maps with a large number of channels have reduced spatial dimensions, thus reducing computational complexity without compromising performance.

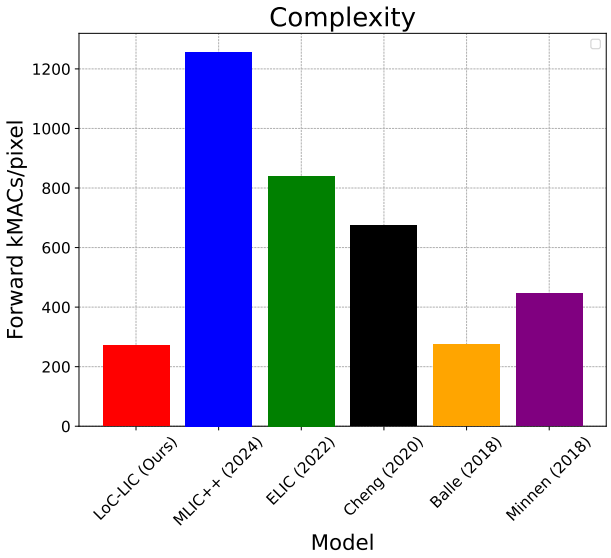


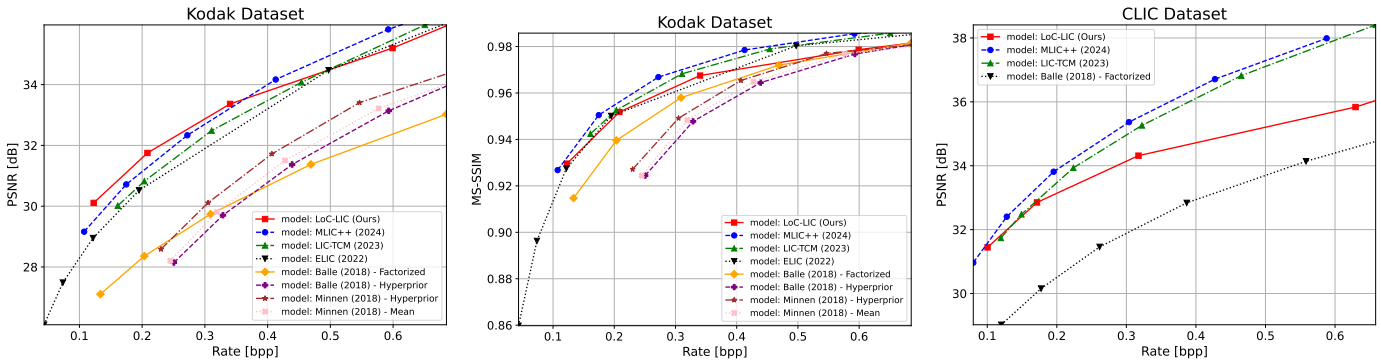
Fig. 3: Comparison of forward complexity between our approach and various learned image compression models

IV. Experiments

We train our models on 256×256 randomly cropped images from a custom dataset containing around 10^6. Our custom dataset images is selected from ImageNet [31] COCO 2017 [32] Vimeo90K [33], and DIV2K [34]. To assess and compare the performance and generalization capability of our model, we conducted validation experiments on two datasets and evaluated its performance against various models. The first dataset utilized is the Kodak dataset [35], a widely adopted benchmark for validating image compression models comprising 24 images. Additionally, we selected the CLIC Professional Valid 2020 dataset [36], which contains 41 high-resolution images, making it well-suited for evaluating compression in the current era of digital high-resolution imagery. We compared our approach against several learned image compression models, including MLIC++ [3], LIC-TCM [10], ELIC [5], and two variations of Balle's (2018) model, Factorized and Hyperprior [8], two variations of Minnen's (2018) model, Mean and Hyperprior [14], as well as Cheng's (2020) Anchor model [12], were included in the comparison.

A. Quantitative analysis

We evaluated the complexity of our model in terms of forward operations measured as kMAC/Pixel, comparing this against other models to assess its efficiency. Using Cheng 2020 [12] as a baseline, illustrated in Figure 3 (a), our model exhibited a significantly reduced complexity of approximately 270 kMAC/Pixel while maintaining superior performance over the Cheng 2020 model, which has a complexity of 933 kMAC/Pixel. Moreover, our model outperformed those by Balle (2018) and Minnen (2018) at both lower and higher bit-rates; these models utilize two different approaches corresponding to varying levels of complexity. An average model complexity was considered for comparison purposes. On the other hand, MLIC++ (2024) proved more efficient with a complexity of around 1256 kMAC/Pixel that we could not surpass.



(a) (b) (c)

Fig. 4: Assessments and comparisons of image compression models using different metrics and datasets. (a) PSNR scores on the Kodak dataset, (b) MS-SSIM scores on the Kodak dataset, and (c) PSNR scores on the CLIC Professional Valid 2020 dataset.

Additionally, our model achieves competitive results in terms of the Structural Similarity Index Measure (SSIM) metric, indicating higher values that align with reduced complexity, as shown in Figure 3

We conducted a comprehensive evaluation of our approach, specifically focusing on a forward path that is responsible for the image encoding and decoding phases. Our analysis, depicted in Figure 3, compares our method to other advanced learned image compression models. Notably, our model demonstrated the lowest complexity, outperforming leading models such as MILC++.

We plotted rate-distortion curves for both Peak Signal-to-Noise Ratio (PSNR) and Multi-Scale Structural Similarity Index Measure (MS-SSIM) using the Kodak and CLIC Professional Validation 2020 datasets, as illustrated in Figure 4. Our model exhibits behavior comparable to that of high-complexity models such as MLIC ++ (2024) [3] and ELIC (2022), while maintaining significantly lower complexity. In terms of PSNR, our model's performance is shown in Figure 4(a), and for MS-SSIM, the performance is depicted in Figure 4(b). It is observed that our model's performance declines with increasing bit rates in both PSNR and MS-SSIM metrics. For larger images, such as those from the CLIC dataset (Figure 4(c)), our model underperforms compared to other state-of-the-art models. This performance gap can be attributed to training conducted exclusively on 256×256 pixel images. This limitation could potentially be addressed by including a mixture of 256×256 and larger sizes, such as 512×512, during training.

B. Qualitative analysis

In our study, we evaluated the visual performance of our model using two distinct datasets, Kodak and CLIC, as illustrated in Figures 5 and 6. Our findings indicate that our model achieves a performance comparable to the state-of-the-art learned image compression model, like MLIC++ (2024) with a marginal increase in the bit rate while maintaining reduced complexity and preserves competitive performance compared to existing models.

V. Conclusion

In this study, we introduce an innovative image compression model with reduced computational complexity, achieving performance comparable to state-of-the-art models. Our method leverages hierarchical feature extraction transforms to significantly lower complexity while effectively maintaining bit rate reduction. We conducted various comparisons with existing learned image compression models, focusing on computational complexity and performance metrics such as PSNR, and LPIPS. Furthermore, we presented our rate-distortion curves with respect to PSNR and MS-SSIM across two benchmark datasets. We also performed qualitative analysis and visual assessments of the compressed images. Our model demonstrated performance on par with state of the art models while retaining minimal complexity.

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] 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).

[5] 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).

Fig. 5: Our novel approach performance compared to MLIC++ and LIC-TCM on image num. 3 from the CLIC Professional Valid 2020 dataset.

Fig. 6: Comparison between our approach and different models on image num. 7 from the Kodak dataset.

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

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

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

[9] 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.

[10] 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.

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

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

[13] 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.

[14] 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.

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

[16] M. Garnelo, D. Rosenbaum, C. Maddison, *et al.*, "Conditional Neural Processes," in *Proceedings of the 35th International Conference on Machine Learning*, Jul. 3, 2018, pp. 1704–1713.

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

[18] J. Gordon, W. P. Bruinsma, A. Y. K. Foong, J. Requeima, Y. Dubois, and R. E. Turner. "Convolutional Conditional Neural Processes." (Jun. 25, 2020).

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

[20] 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.

[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] 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.
[23] S. Lee, J.-B. Jeong, and E.-S. Ryu, "Entropy-Constrained Implicit Neural Representations for Deep Image Compression" vol. 30, pp. 663–667, 2023.
[24] 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.
[25] T. Blard, T. Ladune, P. Philippe, G. Clare, X. Jiang, and O. Déforges. "Overfitted image coding at reduced complexity." (Mar. 18, 2024).
[26] Y. Xu, Y. Shen, J. Zhu, C. Yang, and B. Zhou, "Generative Hierarchical Features From Synthesizing Images," presented at the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 4432–4442.
[27] S. Benyahia, B. Meftah, and O. Lézoray, "Multi-Features Extraction Based on Deep Learning for Skin Lesion Classification" vol. 74, p. 101 701, Feb. 1, 2022.
[28] X. Zhu, Y. Huang, X. Wang, and R. Wang, "Emotion Recognition Based on Brain-like Multimodal Hierarchical Perception" vol. 83, pp. 56 039–56 057, May 1, 2024.
[29] J. Wang, Y. Zou, and H. Wu, "Image Super-Resolution Method Based on Attention Aggregation Hierarchy Feature" vol. 40, pp. 2655–2666, Apr. 1, 2024.
[30] A. Meyer, S. Prativadibhayankaram, and A. Kaup, "Efficient Learned Wavelet Image and Video Coding," in *2024 IEEE International Conference on Image Processing (ICIP)*, Oct. 27, 2024, pp. 1753–1759.
[31] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2009, pp. 248–255.
[32] T.-Y. Lin, M. Maire, S. Belongie, *et al.* "Microsoft COCO: Common Objects in Context." (Feb. 21, 2015).
[33] T. Xue, B. Chen, J. Wu, D. Wei, and W. T. Freeman, "Video Enhancement with Task-Oriented Flow" vol. 127, pp. 1106–1125, Aug. 2019.
[34] E. Agustsson and R. Timofte, "NTIRE 2017 Challenge on Single Image Super-Resolution: Dataset and Study," in *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Jul. 2017, pp. 1122–1131.
[35] Kodak, *E. Kodak lossless true color image suite*, 1993.
[36] "CLIC · Challenge on Learned Image Compression." ().