OpenDIC: An Open-Source Library and Performance Evaluation for Deep-learning-based Image Compression

Wei Gao*
gaowei262@pku.edu.cn
Peking University Shenzhen Graduate School
Peng Cheng Laboratory
Shenzhen, China

Huiming Zheng
hmzheng@stu.pku.edu.cn
Peking University Shenzhen Graduate School
Peng Cheng Laboratory
Shenzhen, China

Chenhao Zhang
chenhaozhang@stu.pku.edu.cn
Peking University Shenzhen Graduate School
Shenzhen, China

Kaiyu Zheng
2301212808@stu.pku.edu.cn
Peking University Shenzhen Graduate School
Shenzhen, China

Zhuozhen Yu
yuzhuozhen@stu.pku.edu.cn
Peking University Shenzhen Graduate School
Shenzhen, China

Yuan Li
liyuan200061@stu.pku.edu.cn
Peking University Shenzhen Graduate School
Shenzhen, China

Hua Ye
yeh@pcl.ac.cn
Peng Cheng Laboratory
Shenzhen, China

Yongchi Zhang
zhangych02@pcl.ac.cn
Peng Cheng Laboratory
Shenzhen, China

Abstract

Deep learning technologies have been popular in the image compression field for some time. An increasing number of deep-learning-based models are proposed to improve Rate-Distortion (RD) performance. Previous algorithms are implemented in the specific platform and can not be applied in cross-platform environments. In this paper, we present an open-source algorithm library called OpenDIC, which integrates a variety of end-to-end image compression methods in cross-platform environments. The contribution and details of the algorithms used in the library are described. To evaluate the performance of these algorithms, we conduct a comprehensive performance test. We compare and analyze each algorithm according to RD performance, running time, and GPU memory occupancy. The algorithm library has been released at https://openi.pcl.ac.cn/OpenDIC/.

*Corresponding author: Wei Gao. This work was supported by The Major Key Project of PCL (PCL2024A02), Natural Science Foundation of China (62271013, 62031013), Guangdong Province Pearl River Talent Program (2021QN020708), Guangdong Basic and Applied Basic Research Foundation (2024A1515010155), Shenzhen Science and Technology Program (JCYJ20230807120808017), and Sponsored by CAAI-MindSpore Open Fund, developed on OpenI Community (CAAIXSJLJJ-2023-MindSpore07).

MM '24, October 28-November 1, 2024, Melbourne, VIC, Australia

ACM ISBN 979-8-4007-0686-8/24/10
https://doi.org/10.1145/3664647.3685519

CCS Concepts

• **Computing methodologies** → **Neural networks**; • **Software and its engineering** → **Open source model**; • **Theory of computation** → **Data compression**.

Keywords

Image Compression, Open-source Model, Algorithm Library, Deep Learning, Cross-platform

ACM Reference Format:
Wei Gao, Huiming Zheng, Chenhao Zhang, Kaiyu Zheng, Zhuozhen Yu, Yuan Li, Hua Ye, and Yongchi Zhang. 2024. OpenDIC: An Open-Source Library and Performance Evaluation for Deep-learning-based Image Compression. In *Proceedings of the 32nd ACM International Conference on Multimedia (MM '24), October 28-November 1, 2024, Melbourne, VIC, Australia.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3664647.3685519

1 Introduction

Deep-learning-based image compression has garnered significant attention in recent years due to its remarkable potential in various applications, including image storage, transmission, and processing. Although traditional compression methods have certain advantages in terms of processing speed [11], there is a significant gap in RD performance. By harnessing the power of neural networks, the advanced methods offer several advantages over traditional methods. They can adaptively learn complex representations from the image data, capture intricate structures, and exploit redundancies, resulting in improved compression efficiency and superior reconstructed image quality. In addition, learning-based compression methods also hold significant advantages in complexity optimization [17] and multimodal processing [22].

Deep-learning-based image compression methods [5, 8, 12, 16, 21] typically involves an encoder-decoder architecture. The encoder network compresses the input image into a high-dimensional latent representation, effectively capturing the essential information of the image. On the other hand, the decoder network reconstructs the image from the compressed representation. The encoder and decoder are trained jointly to optimize the Rate-Distortion (RD) loss. Furthermore, recent developments in deep-learning-based image compression have shown remarkable advancements, particularly in the development of Variational Autoencoder (VAE) [14] architectures. Modifications to the traditional VAE architecture have been proposed to enhance its compression efficiency, which allows VAEs to capture more complex dependencies within image data and generate compressed representations, better preserving the visual details and structures of the original images.

Currently, the ease of use and compatibility of deep learning-based image compression are significant factors limiting its large-scale application. Despite the emergence of numerous new methods in the field, there is a lack of an effective algorithm library for multiple platforms and different testing environments. CompressAI [6] is a landmark open-source project for deep learning image compression, but it can only be used on the PyTorch platform. To address this, we have developed the first cross-platform implementation of an image compression algorithm library, inspired by OpenPointCloud [10] and OpenDMC [9].

In general, our contributions can be summarized as follows:

- First, we provide an algorithm library of deep-learning-based image compression methods for cross-platform implementation, including basic operators, critical functions, and general architectures.
- Second, we categorize the algorithms in the library and analyze the contributions and technical details of these classic algorithms.
- Third, we refine a series of pre-training networks and offer the results of RD performance, running time, and GPU memory occupancy under cross-platform environments.

2 Supported Algorithm Library

The supported algorithms in the library are listed as follows:

- Three Hyperprior-free entropy model methods including E2e_gdn [4], Cae [19] and Iwave [15].
- Two Hyperprior entropy model methods including Hyperprior [5] and Coarse2fine [13].
- Two Hyperprior and context entropy model approaches including Cheng2020 [8] and Checkboard [12].

2.1 Hyperprior-Free Entropy Model Methods

E2e_gdn is a pioneering work in learning-based image compression field, solving many fundamental challenges, including solving the gradient disappearance caused by quantization and network architecture, etc. Its main contributions are listed as follows:

- It proposes a deep-learning-based image compression network consisting of a single linear convolutional layer and local gain control GDN [3].
- It solves the problem of gradient vanishing brought about by quantization by adding uniform noise during training.

Cae is another fundamental work most closely related to E2e_gdn. The differences between the two methods are as follows:

- Cae only replaces the quantization function with a smooth approximation in backward propagation, which means that quantization is still performed as usual in the forward pass.
- It uses a continuous, differentiable function to approximate the discrete probability function and estimates an upper bound on the entropy based on Jensen's inequality [18].

Different from methods above that non-linearly transform images into latent representations leveraging CNNs. Iwave converts images to wavelet-like coefficients. Subsequently, these coefficients are optionally quantized and encoded into bits. The contributions of Iwave are summarized as follows:

- It utilizes a trainable wavelet-like transform to convert images into a series of coefficients without information loss.
- Different from previous schemes, it is a versatile model that supports lossy compression and lossless compression simultaneously.

2.2 Hyperprior Entropy Model Methods

Hyperprior extends the E2e_gdn with a Hyperprior network to capture the spatial redundancy between feature maps. The Hyperprior provides side information for the latent representations, indicating that spatially adjacent elements of the latent representations tend to vary in scale. Coarse2fine inherits Hyperprior structure and extends to dual-level Hyperprior. The contributions of Coarse2fine are listed as follows:

- A dual-level Hyperprior framework is proposed to model the spatial dependencies in the latent representation.
- It utilizes a signal-preserving hyper transform for the analysis and synthesis transforms of the hyper representations

2.3 Hyperprior and Context Entropy Model Methods

Cheng2020 integrates Hyperprior and the context model in the network to predict the probability of latent representations more accurately. The advantages are presented as follows:

- It improves the accuracy and flexibility of the entropy model by leveraging discretized Gaussian mixture likelihoods.
- It embeds self-attention modules into the network architecture to achieve bit allocation, which causes the model to pay more attention to complex texture regions, enhancing the overall RD performance.

Checkboard analyzes the defects of previous context models and makes improvements. Its contributions are listed as follows:

- It proposes a two-stage context prediction pipeline, making context model parallelization possible.
- Compared with mask convolution context model, Checkboard dramatically accelerates decoding speed by nearly 50 times without performance loss.

3 Benchmarking Results and Analysis

3.1 Datasets and Evaluation Metrics

In order to perform uniform testing for the algorithms in OpenDIC, we select Kodak [1] and clic2020-professional-test dataset [2] as the

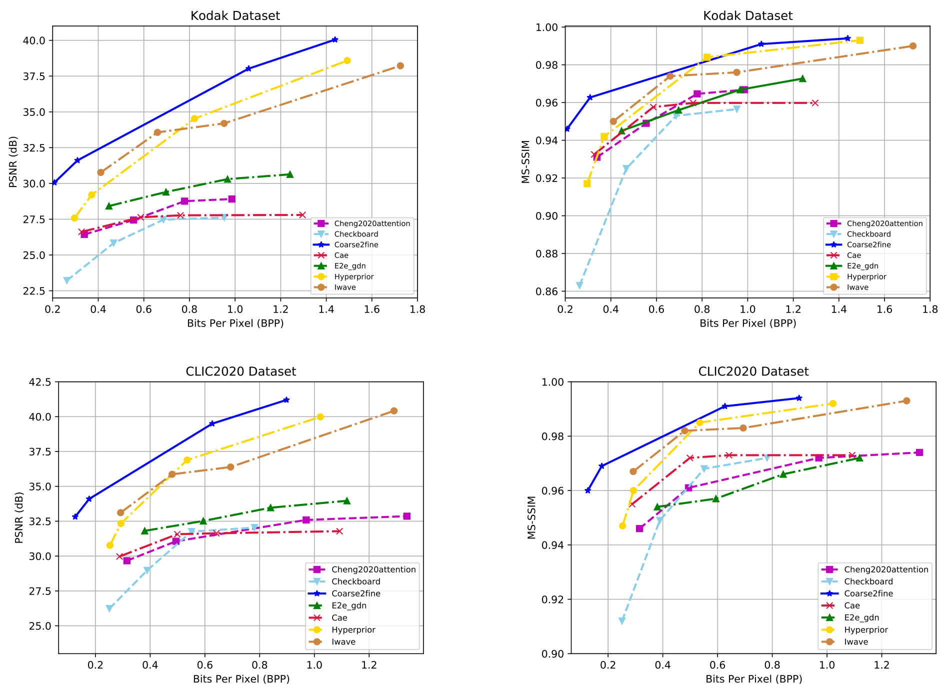

Figure 1: The above two plots represent the Rate-PSNR and Rate-MSSSIM curves tested on the Kodak dataset, while the below two plots show the Rate-PSNR and Rate-MSSSIM curves tested on the clic2020-professional-test dataset.

test dataset. Kodak dataset consists of 24 color images with a resolution of 768×512. The clic2020-professional-test dataset contains more than 200 high-resolution color images. PSNR (Peak Signal-to-Noise Ratio) and MS-SSIM (Multi-scale Structural Similarity) [20] are applied for the Rate-Distortion Performance Evaluation. PSNR is defined based on Mean Square Error (MSE). MS-SSIM measures the distortion level between two images from the perspective of spatial structure similarity. In addition, running time and GPU occupancy are used to measure time complexity and space complexity.

3.2 Rate-Distortion Performance Evaluation

In this subsection, we compare the algorithms in the library, including E2e_gdn, Cae, Hyperprior, Iwave, Checkboard, Cheng2020, and Coarse2fine. Fig. 1 shows the Rate-Distortion (RD) curve results on both Kodak dataset and clic2020-professional-test dataset.

In qualitative analysis, we can find that Coarse2fine achieves the best RD performance. Hyperprior and Iwave rank second and third, respectively. In addition, the RD performances of these three algorithms are significantly better than other algorithms in the library. Subsequently, we calculate BD-PSNR and BD-MSSSIM [7] between other algorithms and Coarse2fine for accurately comparing the RD performance of each algorithm. The results are given in Table 1. Regarding PSNR, Hyperprior and Iwave are inferior to Coarse2fine by 2 dB, while the gap between Coarse2fine and other algorithms exceeds 6 dB. Regarding MS-SSIM, the performance of Hyperprior and Iwave is less than 0.01, while other algorithms have a performance loss of more than 0.02. The results indicate that Coarse2fine achieves the best RD performance. The dual-level Hyperprior framework also empowers the ability to better spatial redundancy removing and more precise probability estimation.

3.3 Running Time and GPU Memory Occupancy Evaluation

In this subsection, we compare the running time and GPU occupancy of all algorithms in the library. We evaluate all the algorithms in Kodak dataset and clic2020-professional-test dataset. We average the running time for each image in two datasets. The running time evaluation is presented in Fig. 2. We can learn that E2e_gdn achieves the best running time. Hyperprior is slower than E2e_gdn due to its Hyperprior framework and gets second place. It is obvious that the running time of the above two algorithms is on the same order of magnitude. Cae, Checkboard, and Cheng2020 reach similar running time. Among them, the running time of Cae is shorter because of its lightweight RNN architecture. Cheng2020 adopts the residual structure which costs a great deal of time. Checkboard adapts the checkboard context model to achieve faster running time than Cheng2020. The running time of Coarse2fine and Iwave is much slower owing to the huge network framework and the complex module design.

The GPU memory evaluation is presented in Fig. 2. It is shown that the GPU memory usage of E2e_gdn, Cae, Hyperprior, Checkboard, and Iwave is less than 6000 MB. Nevertheless, the GPU memory occupancy of Coarse2fine and Cheng2020 significantly exceeds that of other algorithms due to deeper and more complicated network architectures.

Table 1: BD-PSNR and BD-MSSSIM comparisons with Coarse2fine [13] on Kodak dataset and clic2020-professional-test dataset.

Metric \ Algorithm		E2e_gdn	Cae	Hyperprior	Iwave	Checkboard	Cheng2020
BD-PSNR (dB) ↑	Kodak	-6.42	-7.77	-2.38	-2.49	-8.00	-6.98
	Clic2020	-6.63	-7.24	-2.45	-2.89	-8.15	-7.56
	Average	-6.52	-7.50	-2.41	-2.69	-8.07	-7.27
BD-MSSSIM ↑	Kodak	-0.0267	-0.0272	-0.0117	-0.0130	-0.0519	-0.0295
	Clic2020	-0.0318	-0.0186	-0.0089	-0.0079	-0.0338	-0.0272
	Average	-0.0292	-0.0229	-0.0103	-0.0104	-0.0428	-0.0283

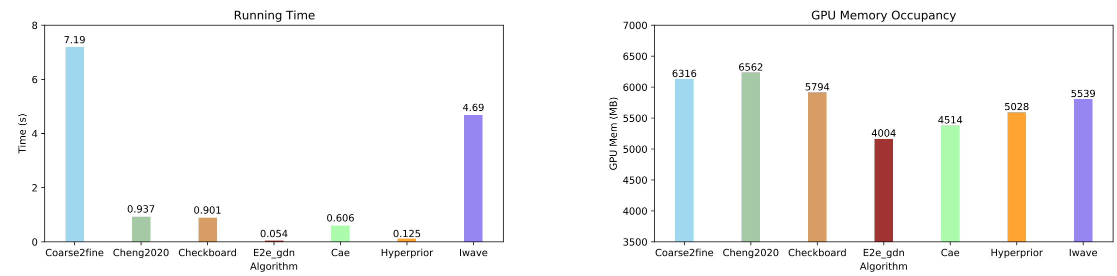

Figure 2: Comparison of Running Time and GPU memory occupancy results.

4 Conclusion

In this paper, we release the deep-learning-based open-source algorithm library for image compression in cross-platform environments. We first briefly introduce the algorithms adopted in the library and divide them into three kinds according to the entropy model. Design principles and characteristics of the model are also illustrated. Subsequently, a performance benchmarking test is established for the classic deep-learning-based image compression algorithms on two datasets. We exhaustively analyze the performance of each model in different tests and the effect of different model designs. We believe that the release of OpenDIC can obviously improve the usability of deep-learning-based image compression models in cross-platform environments.

References

[1] 1997. Kodak Lossless True Color Image Suite. http://r0k.us/graphics/kodak/.
[2] 2020. CLIC 2020 Professional Dataset. https://www.compression.cc/.
[3] Johannes Ballé, Valero Laparra, and Eero P Simoncelli. 2015. Density modeling of images using a generalized normalization transformation. *arXiv preprint arXiv:1511.06281* (2015).
[4] Johannes Ballé, Valero Laparra, and Eero P Simoncelli. 2016. End-to-end optimized image compression. *arXiv preprint arXiv:1611.01704* (2016).
[5] Johannes Ballé, David Minnen, Saurabh Singh, Sung Jin Hwang, and Nick Johnston. 2018. Variational image compression with a scale hyperprior. *arXiv preprint arXiv:1802.01436* (2018).
[6] Jean Bégaint, Fabien Racapé, Simon Feltman, and Akshay Pushparaja. 2020. Compressai: a pytorch library and evaluation platform for end-to-end compression research. *arXiv preprint arXiv:2011.03029* (2020).
[7] G. Bjøntegaard. 2001. Calculation of average PSNR differences between RD curves. *ITU-T SG16 / Q6 Doc. VCEG-M33* (2001).
[8] Zhengxue Cheng, Heming Sun, Masaru Takeuchi, and Jiro Katto. 2020. Learned image compression with discretized gaussian mixture likelihoods and attention modules. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7939–7948.
[9] Wei Gao, Shangkun Sun, Huiming Zheng, Yuyang Wu, Hua Ye, and Yongchi Zhang. 2023. OpenDMC: An open-source library and performance evaluation for deep-learning-based multi-frame compression. In *Proceedings of the 31st ACM International Conference on Multimedia*. 9685–9688.
[10] Wei Gao, Hua Ye, Ge Li, Huiming Zheng, Yuyang Wu, and Liang Xie. 2022. Openpointcloud: An open-source algorithm library of deep learning based point cloud compression. In *Proceedings of the 30th ACM international conference on multimedia*. 7347–7350.
[11] Wei Gao, Hang Yuan, Ge Li, Zhu Li, and Hui Yuan. 2023. Low complexity coding unit decision for video-based point cloud compression. *IEEE Transactions on Image Processing* 33 (2023), 149–162.
[12] Dailan He, Yaoyan Zheng, Baocheng Sun, Yan Wang, and Hongwei Qin. 2021. Checkerboard context model for efficient learned image compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 14771–14780.
[13] Yueyu Hu, Wenhan Yang, and Jiaying Liu. 2020. Coarse-to-fine hyper-prior modeling for learned image compression. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 11013–11020.
[14] Diederik P Kingma and Max Welling. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114* (2013).
[15] Haichuan Ma, Dong Liu, Ning Yan, Houqiang Li, and Feng Wu. 2020. End-to-end optimized versatile image compression with wavelet-like transform. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44, 3 (2020), 1247–1263.
[16] David Minnen, Johannes Ballé, and George D Toderici. 2018. Joint autoregressive and hierarchical priors for learned image compression. *Advances in neural information processing systems* 31 (2018).
[17] Lvfang Tao, Wei Gao, Ge Li, and Chenhao Zhang. 2023. Adanic: Towards practical neural image compression via dynamic transform routing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 16879–16888.
[18] Lucas Theis, Aäron van den Oord, and Matthias Bethge. 2015. A note on the evaluation of generative models. *arXiv preprint arXiv:1511.01844* (2015).
[19] Lucas Theis, Wenzhe Shi, Andrew Cunningham, and Ferenc Huszár. 2017. Lossy image compression with compressive autoencoders. *arXiv preprint arXiv:1703.00395* (2017).
[20] Zhou Wang, Eero P Simoncelli, and Alan C Bovik. 2003. Multiscale structural similarity for image quality assessment. In *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, Vol. 2. Ieee, 1398–1402.
[21] Yuyang Wu, Zhiyang Qi, Huiming Zheng, Lvfang Tao, and Wei Gao. 2021. Deep image compression with latent optimization and piece-wise quantization approximation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 1926–1930.
[22] Huiming Zheng and Wei Gao. 2024. End-to-end rgb-d image compression via exploiting channel-modality redundancy. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 38. 7562–7570.