CATEGORY-DEPENDENT LEARNED IMAGE COMPRESSION FOR SMARTPHONE PHOTOGRAPHY WITH STANDARD-COMPLIANT DECODERS

Abdellah El Mennaoui*† *Ghalia Hemrit** *Jean Luc Dugelay*†

* Huawei Technologies, † EURECOM

ABSTRACT

In this paper, we address the need for tailored image compression by focusing on the specific use case of smartphone photography, particularly selfie, food and landscape images, which dominate user-captured photos. We adapt SegPIC (Segmentation Prior Image Compression) by fine-tuning it individually on a dedicated selfie, food and landscape dataset while keeping the decoder unchanged to maintain compatibility with JPEG AI standard decoding requirements. The model's performance was evaluated using Kodak dataset and the JPEG AI test set, with comparisons based on PSNR and MS-SSIM metrics. To assess the effectiveness of category-specific fine-tuning method, we evaluated the MBT model (Minnen's et al.'s) across the same categories. Our results demonstrate that this fine-tuning improves compression efficiency and image quality compared to training on general datasets (achieving up to 8.2 % BD-rate reduction compared to training on general datasets), highlighting the benefits of category-specific training within standardized frameworks.

Index Terms— image compression, encoder fine-tuning, smartphone photography.

1. INTRODUCTION

The proliferation of smartphones equipped with high resolution cameras has led to an exponential increase in the volume of images captured and shared daily. Efficient image compression is essential to reduce storage requirements and transmission bandwidth, particularly for mobile devices with limited resources. Smartphone photography represents a significant and growing portion of global image capture, driven by the ubiquity of mobile devices and the ease of sharing on social media platforms. Optimizing image compression in this context is critical not only for reducing storage and bandwidth demands but also for enhancing user experience by preserving image quality across diverse mobile applications.

Traditional image compression standards, such as JPEG [1], JPEG 2000 [2], VVC [3], and BPG [4], have been widely adopted but are approaching their performance limits in terms of compression efficiency and visual quality. These methods incorporate modules for transform, quantization, and entropy coding. The transform process focuses on converting images into a more compact set of coefficients by reducing redundancy and eliminating pixel correlations as much as possible. In addition, quantization reduces the precision of these coefficients based on perceptual or statistical criteria, effectively lowering the bit-rate with minimal impact on visual quality, while entropy coding exploits the statistical distribution of the quantized coefficients to further compress the data into a compact bitstream.

In recent years, learning-based image compression (LIC) methods have emerged as a promising alternative, leveraging deep neural networks to achieve superior compression performance. However, most of these models are trained on general-purpose datasets, which may not optimally integrate the specific characteristics of images captured by smartphone cameras. Consequently, there is a need for models that can adapt to the unique features of smartphone images, ensuring better compression efficiency and image quality. The three most popular image categories on social media are studied in this work, including selfie, food and landscape [5].

This paper builds upon our previous work [6] and addresses the need for tailored image compression by proposing a fine-tuning approach for a recent LIC model focused on smartphone-centric categories. We would like to emphasize that our proposed method modifies only the encoder weights; the decoder architecture and bitstream syntax stay fully compliant with the original JPEG AI vision. While this adaptation improves compression efficiency, the decoder remains unchanged to ensure continued compatibility with the JPEG AI standard decoding requirements. The main challenge, however, is not merely to comply with JPEG AI standards [7], but to continue using the same decoder that was originally designed for general purposes, ensuring that specific image categories can still be efficiently compressed without sacrificing the compatibility and functionality of the original decoder.

2. BACKGROUND AND RELATED WORK

Learning-based image compression (LIC) methods have shown remarkable progress in recent years, outperforming traditional approaches by optimizing the rate-distortion trade-off. Notable examples include the work of Ballé *et al.* [8], who introduced a variational autoencoder (VAE) framework for image compression, and Minnen *et al.* [9], who proposed joint autoregressive and hierarchical priors for entropy modelling. More recently, Liu *et al.* [10] introduced the SegPIC model, a segmentation-prior-guided framework that uses class-agnostic masks to achieve superior performance in pixel-fidelity metrics for learned image compression.

Most LIC models are trained on general-purpose datasets like COCO [11] and ImageNet [12], which consist of diverse images from various categories. While this broad training can generalize across different image types, it may not optimally capture the specific characteristics of images taken by smartphones, particularly selfie, food and landscape images. Despite the prevalence of mobile photography, there is a notable lack of LIC methods specifically tailored for the dominant content types captured by mobile devices. This gap underscores the need for category-specific compression techniques that can efficiently handle the images commonly taken by smartphone users

Efforts toward standardizing LIC, such as the JPEG AI initiative [7], have also emerged, focusing on developing models that offer state-of-the-art compression efficiency while ensuring compatibility and inter-operability across devices and platforms, including mobile environments. Testing models on the JPEG AI test set is essential for aligning with these standardization efforts and ensuring fair comparisons among different approaches.

3. METHODOLOGY

3.1. SegPIC Model: Overview and Architecture

The SegPIC model, proposed by Liu *et al.* [10], is a state-of-the-art learned image compression framework that leverages segmentation priors for region-specific compression. It introduces two key modules: the Region-Adaptive Transform (RAT) and the Scale Affine Layer (SAL), which adaptively process different regions of an image based on semantic content. This capability makes SegPIC particularly well-suited for smartphone photography use cases, such as selfies, food, and landscape images, where semantic information plays a critical role in effective compression.

Figure 1 illustrates the overall architecture of the SegPIC framework. The design integrates essential modules, including RAT, SAL, the Window Attention Module (WAM), the Channel-wise Auto-Regressive Model (ChARM), the Factorized Model (FM), and Generalized Divisive Normalization (GDN). The architecture demonstrates how RAT and SAL guide region-specific transformations and enrich contextual information within the encoder and decoder, enabling the model to capture high-level semantic features and enhance image reconstruction quality. Additionally, Downsample and Upsample Blocks with specialized convolutional and transposed convolutional layers facilitate encoding, decoding, and prototype extraction processes.

The RAT module uses class-agnostic segmentation masks to guide region-specific transformations without relying on specific category labels. This flexibility enables the model to learn compression-friendly semantic priors, making it robust for diverse image content. Meanwhile, the SAL module enhances contextual feature representation, contributing to improved compression efficiency and image quality.

3.2. Freezing the Decoder During Fine-Tuning

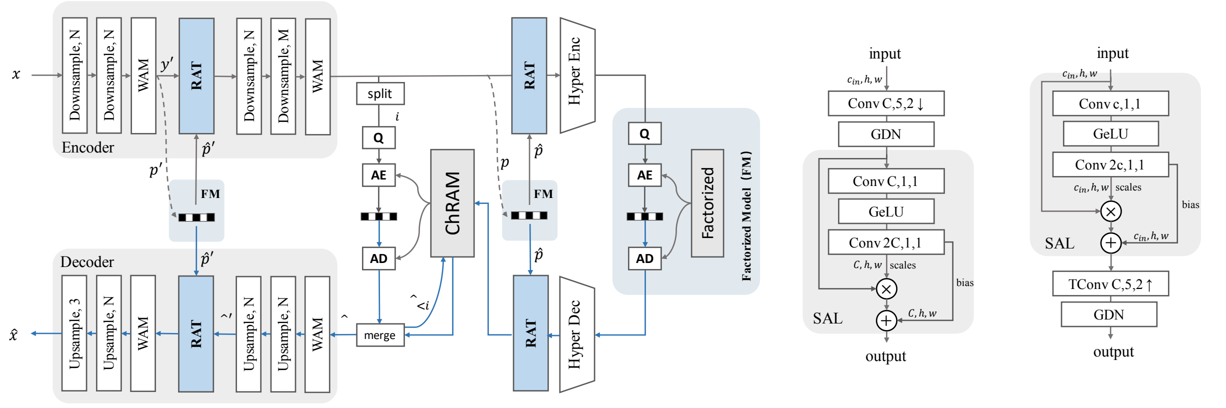
In our approach, we fine-tune only the encoder of the SegPIC model for different image categories such as selfie, food and landscape. The decoder remains unchanged across all rate-distortion trade-offs, as it is fixed from the initial checkpoint. This approach ensures compliance with JPEG AI standardization requirements [13], where the decoder must be standardized and not retrained. By freezing the decoder, we guarantee that the encoded bitstreams produced by the fine-tuned (FT) encoder remain compatible with a fixed decoder, avoiding the need for additional bitrate or computation. This approach demonstrates that, for each category, the encoder is tailored while the decoder remains constant, ensuring practical applicability without modifications to the decoding process. Only the encoder's RAT, SAL, and entropy modules are fine-tuned; the decoder remains fixed for JPEG AI compliance.

3.3. Handling Inference Without Masks

Segmentation masks are treated as privileged information during the training phase, helping the encoder to learn more effective compression strategies by guiding the model's attention to specific regions of the image. However, for inference, these masks are not available, especially for datasets like Selfie [14][15], Food101 [16] and landscape dataset [17], where segmentation masks are not provided. To address this, we use 4×4 grid partitions (aligned with the codec's native 4×4 transform blocks, requiring no side-information or decoder changes) during the inference phase to replace segmentation masks. The fixed decoder effectively uses these grid partitions, demonstrating that the model has learned to generalize and capture relevant contextual information even without access to the detailed segmentation data used during training. This approach ensures that the compression remains efficient and maintains high reconstruction quality, even when explicit mask information is unavailable.

3.4. Training Setup

The fine-tuning process involved training the encoder separately on smartphone selfie, food, and landscape image datasets, keeping the decoder unmodified to ensure compatibility with standardized decoders. We optimized the model using a rate-distortion trade-off loss function, focusing on minimizing bit per pixel (bpp) while maximizing image quality, as measured by Peak Signal-to-Noise Ratio (PSNR) and



(a) The overall framework (b) Downsample block (c) Upsample block

Fig. 1. Framework of SegPIC, from [10]

Multi-Scale Structural Similarity Index Measure (MS-SSIM). The training strategy involved adjusting hyperparameters such as the learning rate, number of epochs, and batch size to balance compression efficiency and image quality. Detailed training settings are provided in the following section.

4. EXPERIMENTS AND RESULTS

4.1. Model Fine-Tuning

To tailor the SegPIC model for specialized image compression tasks, we fine-tuned it using subsets from the Selfie dataset [14][15], Food101 dataset [16], and 10,000 landscape images from the Flickr dataset [17]. Initially, the SegPIC model was trained for 400 epochs on the COCO-Stuff dataset [18], with a batch size of 32 and 32 worker threads. This initial training employed a learning rate of 1×10^{-4}, with a training patience of 16 epochs and a minimum learning rate of 5×10^{-6}. For the fine-tuning phase, the model was further trained for 100 epochs, maintaining a batch size of 32 but reducing the number of workers to 24. We experimented with different λ values (0.0018, 0.0035, 0.0067, 0.0130, 0.0250, and 0.0483), where λ is the Lagrange multiplier controlling the trade-off between bitrate and distortion, using the Mean Squared Error loss, and we reset the model to its pre-trained state before fine-tuning on each dataset independently to avoid any cross-dataset influence. The learning rate was adjusted to facilitate weight updates, with an initial learning rate set to 1×10^{-4}, a patience of 10 epochs, and a minimum learning rate of 5×10^{-5}.

The fine-tuning process for the Selfie dataset [14][15] involved using a subset of 10,000 images, divided into 8,000 for training, 1,000 for validation, and 1,000 for testing. For the Food101 dataset [16], we selected 10,100 images in total, shuffled across all food categories to ensure a diverse representation. These images were split similarly, with 8,080 images used for training, 1,010 for validation, and 1,010 for testing.

The same split was applied to the 10,000 landscape images from the Flickr dataset [17], ensuring balanced representation within each dataset. This approach allowed the model to adapt effectively to the characteristics of selfie, food and landscape images, which are prevalent in smartphone photography.

The SegPIC model comprises 83.5 M parameters in total, of which 16.5 M are trainable during fine-tuning. On a single GPU, encoding a Kodak image (768×512) takes approximately 142 ms and decoding takes 130 ms.

4.2. Results Analysis

The results, as detailed in Table 1, indicate a clear improvement in performance for the fine-tuned SegPIC compared to the pre-trained model when tested on Selfie, Food101, and landscape datasets (see Fig. 2). For both PSNR and MS-SSIM metrics, the fine-tuned model demonstrates superior performance, particularly at lower bit-per-pixel (bpp) values, notably in Figure 3, where maintaining image quality is more challenging. The evaluation was extended to include a subset of 3,931 selfie images from Flickr30k [19], where the fine-tuned SegPIC model showed an improvement over the pre-trained model. We observed a bitrate saving of -3.40% for a fixed PSNR and -5.80% for a fixed MS-SSIM, as indicated by the reduction in the Bjøntegaard rate (BD-rate) [20], which measures the average bitrate change needed to maintain the same quality level relative to a reference. This highlights the model's ability to generalize effectively to new and unseen selfie images, underscoring the advantage of fine-tuning for smartphone image compression.

Additionally, we tested the fine-tuned SegPIC model trained on Food101 on a subset of the Selfie dataset to investigate whether the performance is significantly affected by the image category. As expected, the results revealed a significant drop in performance with a BD-rate increase of +4.83% for a

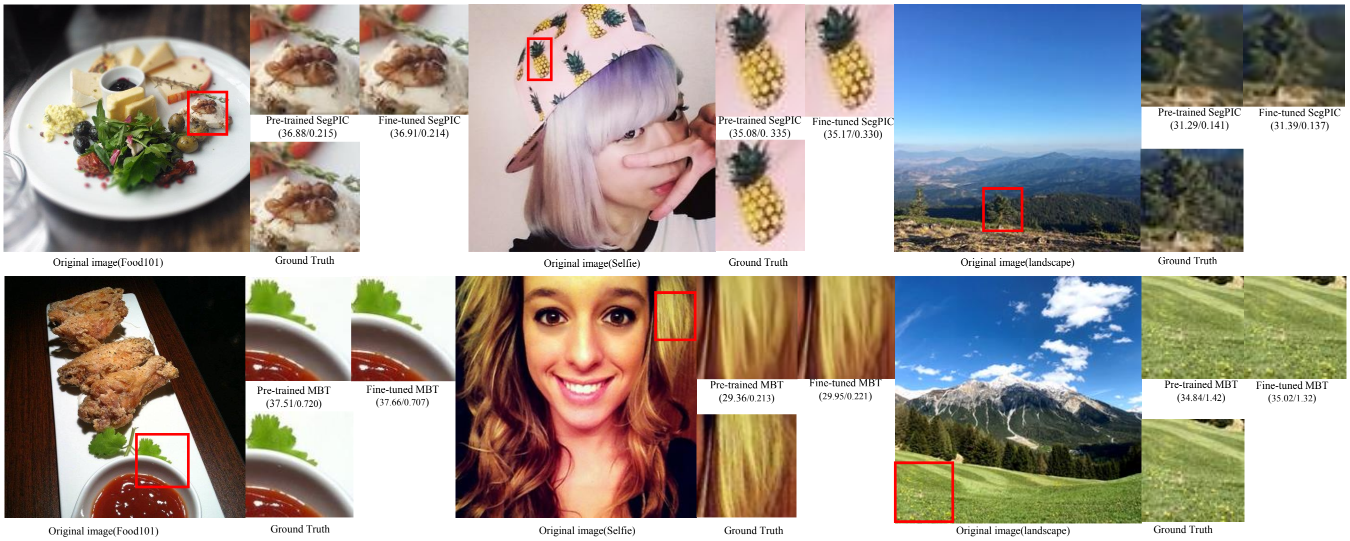


Fig. 2. Visualization of the reconstructed images between the pre-trained and fine-tuned SegPIC and MBT models on Food101, Selfie, and landscape datasets. The metrics shown are (PSNR↑/bpp↓), indicating that our fine-tuned models capture object contours more accurately with less bitrate.

Table 1. Average BD-Rate PSNR/MS-SSIM across different bit rates for SegPIC fine-tuned on different datasets (Relative to Pre-Trained)

Dataset	**FT SegPIC Selfie**		**FT SegPIC Food101**		**FT SegPIC landscape**	
	BD PSNR	**BD MS-SSIM**	**BD PSNR**	**BD MS-SSIM**	**BD PSNR**	**BD MS-SSIM**
Selfie	**-2.69%**	**-2.89%**	**+4.83%**	**+3.13%**	-	-
Flickr30k (selfie)	**-3.40%**	**-5.80%**	-	-	-	-
Food	-	-	**-3.87%**	**-4.02%**	-	-
landscape	-	-	-	-	**-1.13%**	**-1.40%**
Kodak	+0.21%	+0.25%	+0.84%	-0.19%	-0.18%	-0.04%
JPEG AI	**+1.50%**	-0.20%	-0.80%	-0.10%	+0.21%	-0.04%

Table 2. Average BD-Rate PSNR/MS-SSIM across different bit rates for MBT fine-tuned on different datasets (Relative to Pre-Trained)

Dataset	**FT MBT Selfie**		**FT MBT Food101**		**FT MBT landscape**	
	BD PSNR	**BD MS-SSIM**	**BD PSNR**	**BD MS-SSIM**	**BD PSNR**	**BD MS-SSIM**
Selfie	**-7.24%**	**-5.27%**	**-**	**-**	-	-
Flickr30k (selfie)	**-3.29%**	**-4.12%**	-	-	-	-
Food	-	-	**-5.10%**	**-4.82%**	-	-
landscape	-	-	-	-	**-2.38%**	-0.90%
Kodak	-0.65%	+0.30%	**-1.04%**	+0.64%	+0.81%	+0.70%

fixed PSNR and +3.13% for a fixed MS-SSIM. This confirms that while fine-tuning on one domain (e.g., Food101) brings clear benefits for similar images, it can degrade performance on dissimilar domains (e.g., Selfie). It also explains the observed increase in BD-rate when testing on the JPEG AI test set [21], where the pre-trained model, trained on a general dataset, shows better generalization across varied image categories, while fine-tuned models are more specialized.

When it comes to landscape images, the fine-tuned SegPIC model demonstrates promising results as well, achieving a bitrate saving of -1.13% for a fixed PSNR and -1.40% for a fixed MS-SSIM. This shows that the fine-tuned model can generalize to other outdoor photography use cases, highlighting its robustness across various landscape photography scenarios.

4.3. Generalization Analysis

To further evaluate how category-specific fine-tuning generalizes to different architectures, we also tested the MBT model introduced by Minnen *et al.* [9, 22], which integrates joint autoregressive and hierarchical priors within a learned image compression framework for state-of-the-art rate–distortion performance. Table 2 presents the BD-Rate results for MBT models fine-tuned on different datasets. This analysis highlights how dataset-specific fine-tuning impacts performance across various image categories.

The fine-tuned MBT model shows improvements, achieving the highest BD-Rate reduction of -7.24% for a fixed PSNR for the Selfie category and the lowest reduction of -2.38% for the landscape category. Similarly, for MS-SSIM, the fine-tuned MBT model achieves the highest reduction of -5.27% for the Selfie category and the lowest reduction of -0.9% for the landscape category. This confirms that fine-tuning on targeted datasets leads to noticeable gains in compression efficiency across different categories, demonstrating the model's ability to efficiently compress images in various scenarios, and we also observe improvement especially for high bpp values as illustrated in Figure 4, where the model preserves fine details more effectively.

Interestingly, the performance on Kodak dataset [23], a general-purpose benchmark, shows minor variations across all fine-tuned models, with a BD-Rate ranging from +0.65% to -1.04% for a fixed PSNR. This indicates that while fine-tuning on specific datasets leads to improvements in targeted categories, the generalization to unseen data remains stable, with only slight changes in bitrate.

Finally, Figures 5 and 6 show that on both Kodak dataset [23] and the JPEG AI test set [21], fine-tuned SegPIC and MBT models remain nearly unaffected, with BD-rate shifts within ±0.25 % relative to their pre-trained counterparts, confirming that fine-tuning does not impair general dataset performance while providing notable domain-specific gains. As illustrated in Figure 2, the reconstructed outputs of the fine-tuned SegPIC and MBT models on Food101, Selfie, and landscape images visibly preserve object contours with fewer bits than the pre-trained models.

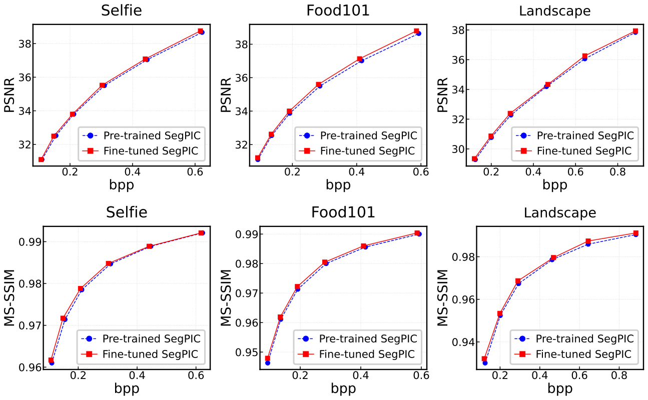


Fig. 3. Performance comparison at different compression rate of pre-trained and fine-tuned SegPIC model on various datasets: PSNR (top row) and MS-SSIM (bottom row) comparisons.

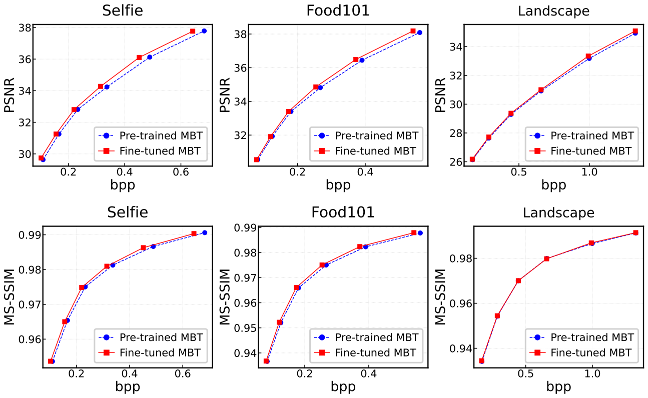


Fig. 4. Performance comparison at different compression rate of pre-trained and fine-tuned MBT model on various datasets: PSNR (top row) and MS-SSIM (bottom row) comparisons.

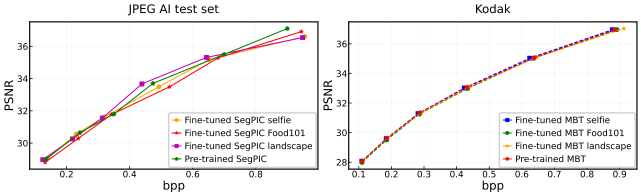


Fig. 5. Performance comparison at different compression rate of pre-trained and fine-tuned SegPIC on JPEG AI test set and MBT models Kodak dataset.

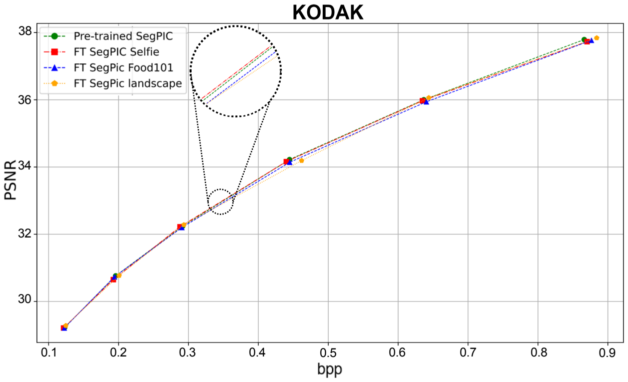


Fig. 6. Performance comparison of pre-trained and fine-tuned SegPIC on Kodak datasets.

5. CONCLUSION AND FUTURE WORK

In this paper, we presented an approach to enhance the performance of learning-based image compression (LIC) models for smartphone-specific use cases, focusing on selfie, food and landscape images. By fine-tuning the SegPIC model, we demonstrated that adapting the encoder while keeping the decoder unchanged can improve compression efficiency and image quality compared to models trained on general datasets. We also showed that same conclusions apply to another LIC model, MBT, when fine-tuned on the smartphone-specific categories.

For future work, we plan to extend our research by exploring the impact of fine-tuning the SegPIC model across a broader range of image categories—such as pets and portrait images—to validate its adaptability for various smartphone photography scenarios while ensuring that the model delivers optimal quality, speed, and resource efficiency through real-time compression performance evaluations on diverse mobile devices. In tandem with these efforts, we will retrain the model on a comprehensive dataset that incorporates representative proportions of each category, ensuring robust performance across all typical smartphone use cases. Moreover, we aim to develop a pre-classification mechanism to dynamically select the most appropriate encoder for each image type, thereby optimizing both compression efficiency and output quality for a wide range of content.

6. REFERENCES

[1] William B Pennebaker and Joan L Mitchell, *JPEG Still Image Data Compression Standard*, Springer-Verlag, New York, NY, USA, 1992.

[2] David S Taubman, Michael W Marcellin, and Majid Rabbani, *JPEG2000 Image Compression Fundamentals, Standards and Practice*, Kluwer Academic, Boston, MA, USA, 2002.

[3] Joint Video Experts Team (JVET), "Versatile video coding," Online. Available: https://jvet.hhi.fraunhofer.de/, April 2021, Accessed: June 2024.

[4] Fabrice Bellard, "Bpg image format," Online. Available: https://bellard.org/bpg/, September 2014, Accessed: June 2024.

[5] Eric Massip, Shintami Chusnul Hidayati, Wen-Huang Cheng, and Kai-Lung Hua, "Exploiting category-specific information for image popularity prediction in social media," in *2018 IEEE International Conference on Multimedia & Expo Workshops*, 2018, pp. 45–46.

[6] Abdellah El Mennaoui, Ghalia Hemrit, and Jean Luc Dugelay, "Optimized image compression for mobile photography," in *Proceedings of the 2025 Data Compression Conference (DCC)*, 2025, p. 364.

[7] Joint Photographic Experts Group (JPEG), "JPEG AI—learning-based image coding standardization," https://jpeg.org/jpegai/, 2021.

[8] Johannes Ballé, Valero Laparra, and Eero P Simoncelli, "End-to-end optimized image compression," in *Proceedings International Conference Learning Representations*, April 2017, pp. 1–27.

[9] David Minnen, Johannes Ballé, and George D Toderici, "Joint autoregressive and hierarchical priors for learned image compression," *Advances in neural information processing systems*, vol. 31, pp. 10771–10780, 2018.

[10] Yuxi Liu, Wenhan Yang, Huihui Bai, Yunchao Wei, and Yao Zhao, "Region-adaptive transform with segmentation prior for image compression," in *European Conference on Computer Vision*, 2024, pp. 181–197.

[11] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick, "Microsoft COCO: Common objects in context," in *Proceedings of the European Conference on Computer Vision*, 2014, pp. 740–755.

[12] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255.

[13] João Ascenso, Elena Alshina, and Touradj Ebrahimi, "The jpeg ai standard: Providing efficient human and machine visual data consumption," *IEEE MultiMedia*, vol. 30, no. 1, pp. 9–17, 2023.

[14] Mahdi M Kalayeh, Misrak Seifu, Wesna LaLanne, and Mubarak Shah, "How to take a good selfie?," in *Proceedings of the ACM Multimedia Conference*, 2015, pp. 923–926.

[15] University of Central Florida, "Selfie dataset," 2015, https://www.crcv.ucf.edu/data/Selfie/.

[16] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool, "Food-101 – mining discriminative components with random forests," in *Proceedings European Conference Computer Vision*, 2014, pp. 446–461.

[17] Bart Thomee, David A Shamma, Gerald Friedland, Benjamin Elizalde, Karl Ni, Douglas Poland, Damian Borth, and Li-Jia Li, "Yfcc100m: The new data in multimedia research," *Communications of the ACM*, vol. 59, no. 2, pp. 64–73, 2016.

[18] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari, "Coco-stuff: Thing and stuff classes in context," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1209–1218.

[19] Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik, "Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 2641–2649.

[20] Gisle Bjontegaard, "Calculation of average psnr differences between rd-curves," *ITU SG16 Doc. VCEG-M33*, 2001.

[21] Joint Photographic Experts Group (JPEG), "JPEG AI Dataset," Online. Available: https://jpeg.org/jpegai/dataset.html, 2022, Accessed: April 2024.

[22] InterDigitalInc, "JPEG AI–learning-based image coding standardization," https://github.com/InterDigitalInc/CompressAI, Accessed: May 2024.

[23] Rich Franzen, "Kodak lossless true color image suite," *source: http://r0k. us/graphics/kodak*, vol. 4, no. 2, pp. 9, 1999.