Architectural Analysis of Hybrid-Attention: 2-Layer BACF-Net for Image Compression

Chen-Lin Chang, Hsu-Feng Hsiao
Department of Computer Science
National Yang Ming Chiao Tung University, Hsinchu, Taiwan

***Abstract*—Learned image compression has increasingly benefited from hybrid architectures that aim to combine the local modeling of CNNs with the global context of transformers. Our prior work, BACF-Net, established a competitive baseline by introducing the Bifurcated Attention-Convolution Fusion (BACF) block to integrate these technologies. In this paper, we advance this framework by proposing an optimized 2-layer BACF-Net, demonstrating that a strategically shallower depth yields a superior rate-distortion trade-off. We support this optimization with a comprehensive architectural analysis that was previously unexplored. Our ablation studies provide empirical validation for the complementary integration of mixed-attention mechanisms, the efficiency of our asymmetric encoder-decoder design, and the critical role of split-attention in outperforming traditional GDN. Our refined model achieves competitive performance, highlighting that rigorous architectural analysis is essential for optimizing hybrid compression models.**

I. Introduction

The relentless growth of visual data has established efficient image compression as a cornerstone of modern data management and transmission. For decades, this field was dominated by hand-crafted codecs, such as JPEG [1], HEVC [2], and the current state-of-the-art, Versatile Video Coding (VVC) [3]. While highly optimized, these traditional methods are fundamentally constrained by fixed, manually engineered transformation and entropy coding pipelines. Learned image compression (LIC) has emerged as a transformative paradigm, leveraging end-to-end optimization of deep neural networks. This data-driven approach allows models to automatically discover more compact and perceptually aligned latent representations, enabling LIC frameworks to consistently surpass the rate-distortion (R-D) performance of VVC [4], [5].

Within the LIC landscape, two dominant architectural paradigms have emerged. On the one hand, Convolutional Neural Networks (CNNs) excel at capturing local spatial correlations and translation invariance, making them highly effective for texture and pattern modeling [4], [6]. However, the inductive bias imposed by their local receptive fields inherently limits their ability to model long-range dependencies. On the other hand, Vision Transformers [7] leverage self-attention mechanisms to capture global contextual relationships across the entire image. While powerful, this global modeling often incurs significant computational complexity and may overlook fine-grained local details. Consequently, hybrid architectures that integrate the complementary strengths of both CNNs and Transformers represent a promising direction for advancing compression performance [8], [9].

Building upon this hybrid trend, our prior work [10] introduced the Bifurcated Attention-Convolution Fusion (BACF) network. The core of this framework is the BACF block, a novel dual-path parallel design: one path employs a residual CNN augmented with Split Attention, while the other utilizes a vision transformer module (instantiated as either MaxViT [11] or Swin Transformer [12]). This parallel architecture is designed to simultaneously process fine-grained local textures and high-level global semantic information before fusing them. The resulting framework established a solid performance baseline, demonstrating competitive rate-distortion results and validating the potential of this fusion strategy.

While BACF-Net [10] demonstrated strong empirical results, its initial presentation focused on the final architecture, leaving the underlying design rationale and the impact of individual components unexplored. Specifically, the efficacy of the combined mixed-attention, asymmetric decoding, and split-attention mechanisms was not empirically validated. Furthermore, the architectural configuration, such as the depth of the core encoder, was presented as a fixed choice, leaving open the question of whether it represented the optimal trade-off between performance and complexity.

This paper addresses these gaps and presents two primary contributions. First, we provide a comprehensive architectural dissection of the BACF-Net framework. We conduct a rigorous series of ablation studies to empirically validate the core design choices, evaluating: (a) the necessity of the mixed-attention strategy over single-transformer variants, (b) the efficiency of the asymmetric decoder, (c) the significant R-D gains yielded by the split-attention module compared to traditional GDN, and (d) the performance equivalence of serial versus parallel connection strategies, validating our simpler cascaded design. Second, informed by this analysis, we propose an architectural optimization. Our investigation into encoder depth reveals that the original 3-layer stack was suboptimal. We introduce a refined 2-layer core encoder that not only reduces complexity but, more importantly, achieves superior rate-distortion performance, establishing a new optimal configuration for the BACF-Net architecture.

II. Methodology: The Optimized BACF-Net

A. Overall Framework

Our proposed architecture, illustrated in Fig. 1, is built upon the widely adopted hyperprior-based framework [6], [13]. An input image x is first transformed by the core encoder g_a into

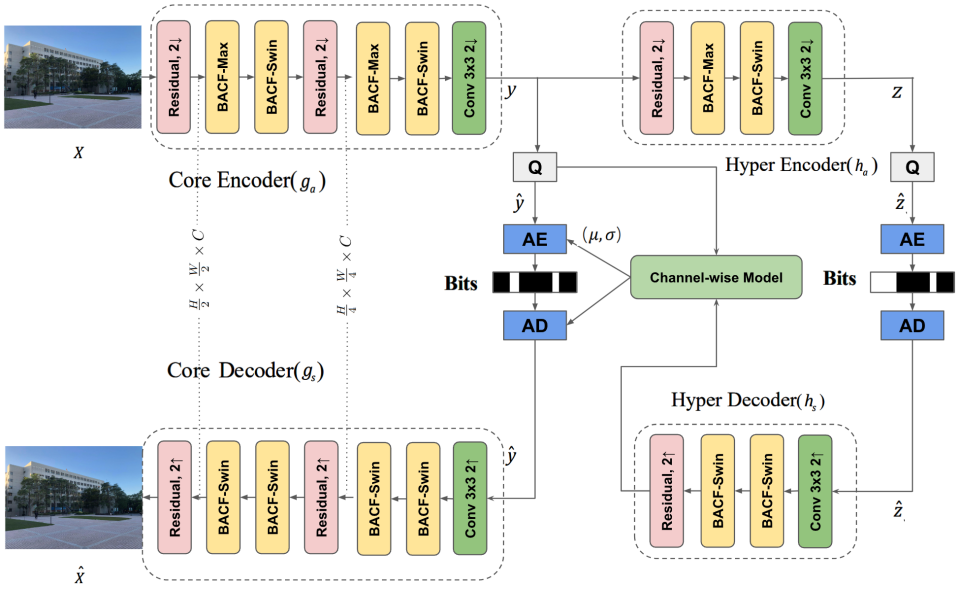

Fig. 1: The overall architecture of our optimized learned image compression framework. The core encoder (g_a) and core decoder (g_s) are refined to use two transformation layers, each composed of a residual block (Fig. 3) and cascaded BACF blocks (Fig. 2). The symbols $2\uparrow$ and $2\downarrow$ represent upsampling and downsampling operations, while Q, AE, and AD denote quantization, arithmetic encoding, and decoding, respectively.

a latent representation y. To capture spatial redundancies and provide side information, y is further analyzed by a hyper-encoder h_a to produce a hyper-latent z. Both y and z are quantized ($\hat{y}$, $\hat{z}$) and compressed into a bitstream using entropy coding. On the decoder side, a hyper-decoder h_s reconstructs the parameters (e.g., mean and scale) of $\hat{y}$'s distribution from $\hat{z}$, enabling accurate probability estimation via a channel-wise entropy model [14]. Finally, the core decoder g_s reconstructs the image $\hat{x}$ from $\hat{y}$. The entire model is optimized end-to-end by minimizing the rate-distortion (R-D) loss:

$$L = R(\hat{y}) + R(\hat{z}) + \lambda \cdot D(x, \hat{x}), \tag{1}$$

where R denotes the estimated bitrates of the latents, D is the distortion between x and $\hat{x}$ (measured by MSE), and λ controls the R-D trade-off.

B. The Bifurcated Attention-Convolution Fusion (BACF) Block

The fundamental component of our framework is the Bifurcated Attention-Convolution Fusion (BACF) block, depicted in Fig. 2. This module is designed to integrate the complementary strengths of Vision Transformers and residual CNNs via a parallel configuration. An input tensor first passes through a 1×1 convolutional layer to unify feature dimensions. The resulting tensor is then split evenly along the channel dimension, bifurcating the features into two distinct paths: (1) a vision transformer branch and (2) a residual CNN branch. This parallel design serves a dual purpose: it reduces the computational load for each subsequent branch and, more critically, it allows the network to independently and simultaneously process local patterns (via CNN) and global dependencies (via Transformer), thereby enhancing its feature extraction capabilities [9].

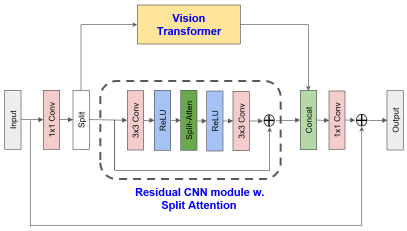

Fig. 2: The Bifurcated Attention-Convolution Fusion (BACF) block. It integrates vision transformers (BACF-Max or BACF-Swin) and a residual CNN module with split attention in a parallel configuration.

The Vision Transformer branch, shown in Fig. 2, processes its half of the feature channels. In our framework, this module is instantiated as two distinct variants, which in turn define two types of BACF blocks. The BACF-Max variant employs a MaxViT [11] block to capture global, multi-axis attention. Conversely, the BACF-Swin variant utilizes a Swin Transformer [12] block for efficient, shifted window-based local attention. As detailed in Section II-C, these two block types are cascaded within the full encoder architecture.

Running in parallel to the vision transformer, the second path is the residual CNN model with split attention. This branch (bottom path of Fig. 2) also processes its half of the feature channels, passing them through a 3×3 convolution followed by a ReLU activation. Critically, this path is enhanced by a ResNeSt-inspired [15] split-attention mechanism

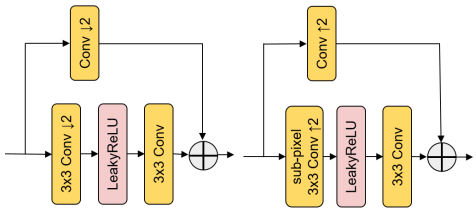

Fig. 3: The residual blocks for spatial downsampling ($2\downarrow$) via strided convolution in the encoder (left) and upsampling ($2\uparrow$) via sub-pixel convolution in the decoder (right).

[16]. This module, which our analysis in Section III proves is important, allows the network to perform feature-map attention across different splits, adaptively refining local representations.

Finally, the outputs from the Vision Transformer path (e.g., MaxViT) and the residual CNN path are fused. The two tensors are concatenated along the channel dimension. A concluding 1×1 convolution then integrates these parallel-processed local and global representations, producing the final output of the BACF block.

C. Optimized 2-Layer Architecture

We present the proposed optimized BACF-Net framework, illustrated in Fig. 1. This architecture integrates the BACF blocks described in Section II-B into a complete compression pipeline. This configuration results from the extensive architectural analysis presented in Section III. Specifically, a key contribution of this architecture is the refinement of the core encoder g_a to an optimal depth of two transformation layers, a direct optimization over the 3-layer stack used in our preliminary work [10].

As shown in Fig. 1, the core encoder g_a is responsible for transforming the input image x into the compact latent representation y. Our optimized design for g_a consists of two sequential transformation layers. Each transformation layer begins with a strided convolutional residual block (Fig. 3, left) that performs spatial downsampling ($2\downarrow$) [4]. This is immediately followed by a cascaded pair of our fusion blocks: first, a BACF-Max block to capture global context, and second, a BACF-Swin block to refine local features. After the input passes through these two transformation layers, a final 3×3 convolution maps the features to the target channel dimension of the latent representation y. This 2-layer design is a deliberate optimization (validated in Section III) that balances deep feature extraction with computational efficiency.

A key feature of our framework is the asymmetric architecture. In contrast to the core encoder's "Max+Swin" hybrid design, the core decoder g_s and the entire hyper-decoder h_s are intentionally made more lightweight. As shown in Fig. 1, their transformation layers utilize only the BACF-Swin blocks. This asymmetric approach, which we validate in Section III, achieves a favorable balance between model capacity and efficiency by leveraging SwinV2's localized attention for high-fidelity reconstruction.

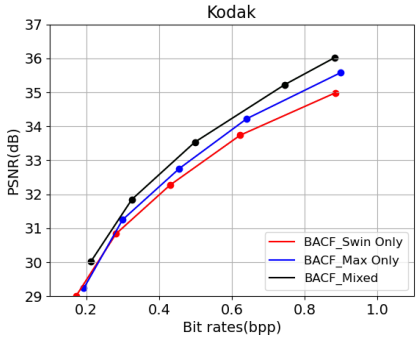

Fig. 4: Comparative analysis of R-D performance on the Kodak dataset for BACF blocks utilizing Swin Transformer V2 only, MaxViT blocks only, and a mixture of both.

III. Architectural Analysis and Ablation

The optimized architecture presented in Section II results from a detailed investigation into the framework's core components. This section details the ablation studies that validate our design choices, providing empirical justification absent in our preliminary work [10]. All models in this section were trained on a 30,000-image subset of OpenImages [17] to ensure a fair and consistent comparison.

A. Necessity of Mixed-Attention in the Encoder

Our first investigation probes the fundamental premise of the "hybrid-attention" BACF encoder. To validate the hypothesis that a mixture of transformer variants is beneficial, we designed three distinct encoder configurations: (i) Swin Only: All BACF blocks in the encoder use only Swin Transformer V2. (ii) MaxViT Only: All BACF blocks in the encoder use only MaxViT. (iii) Mixed: The cascaded BACF-Max $\rightarrow$ BACF-Swin design proposed in Section II-C. These experiments utilized the previous 3-layer stack for a direct comparison of the attention mechanisms themselves.

The results presented in Fig. 4 are unequivocal. The Mixed configuration (black line) achieves a consistently superior rate-distortion curve compared to architectures relying on a single transformer type. This outcome provides strong empirical evidence for our central hypothesis: the attention mechanisms are complementary, rather than redundant. The multi-axis attention in MaxViT facilitates global context aggregation and long-range dependency modeling, while the shifted-window strategy of SwinV2 enables the efficient capture of localized structures. By integrating these distinct mechanisms within our cascaded design, the BACF encoder attains a more balanced and expressive feature representation in the latent space, validating the design's contribution to compression efficiency.

B. Efficacy of Asymmetric Design

Our next analysis investigates the decision to use an asymmetric module configuration within the encoder-decoder framework. In this context, asymmetry refers to the specific transformer variants employed, rather than the layer depth. We

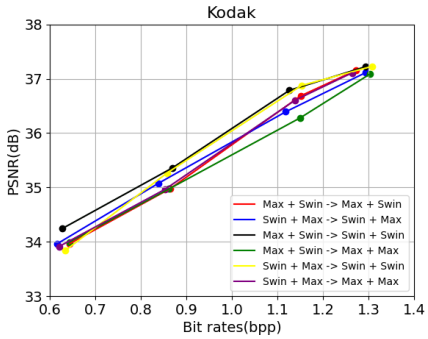

Fig. 5: Rate-distortion performance comparison on the Kodak dataset for various symmetric and asymmetric combinations of vision transformers in the BACF blocks (e.g., Encoder → Decoder).

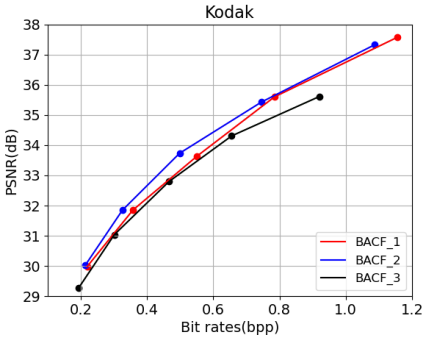

Fig. 6: Rate-distortion performance comparison (PSNR vs. bpp) on the Kodak dataset for models with 1, 2, and 3 stacked BACF transformation layers in the encoder and decoder.

compare symmetric configurations (e.g., identical modules in both encoder and decoder) against our proposed asymmetric design, which utilizes the full "Max+Swin" mixture in the encoder but employs "Swin" blocks exclusively in the decoder. To validate this choice, we evaluated six distinct architectural pairings. For this specific study, all models were configured with the same 2-layer depth (as determined in Section III-C) to isolate the effect of module composition.

The results plotted in Fig. 5 clearly support our design choice. The asymmetric configuration (represented by the black line, `Max+Swin` → `Swin+Swin`) yields the most favorable rate-distortion trade-off among all tested variants. This outcome suggests an efficient architectural balance: the encoder benefits from the powerful feature extraction of the full "Max+Swin" hybrid module to model complex dependencies, while the decoder achieves robust high-fidelity reconstruction using only the more lightweight SwinV2 blocks. This asymmetric module composition provides superior compression performance without incurring the full computational load associated with a symmetric hybrid decoder.

C. Architectural Depth Optimization

Perhaps our most critical investigation concerns the optimal depth of the core encoder and decoder. The performance of deep learning models does not always monotonically improve with depth; adding layers can introduce vanishing gradients, increase parameter counts, and, in some cases, even degrade performance [18]. To identify the "sweet spot" for our BACF framework, we conducted an ablation study varying the number of stacked transformation layers (each containing a residual block and the BACF block pair) in both the encoder and decoder. We specifically examined configurations with one (BACF_1), two (BACF_2), and three (BACF_3) stacked layers.

The results, illustrated in Fig. 6, reveal a clear optimal point. Stacking two BACF layers consistently outperforms the single-layer configuration, confirming that deeper feature fusion is necessary for rich representational power. However, increasing the depth further to three stacked layers, the configuration used in our preliminary work [10], does not yield additional benefits. Instead, it leads to a marginal but consistent performance degradation across the bitrate range. These findings are critical: they demonstrate that a balanced depth of two BACF layers strikes the most effective trade-off between model complexity and compression performance. Therefore, we adopt the 2-layer architecture as the definitive, optimized framework for all subsequent evaluations in this paper.

TABLE I: Comparison of different nonlinear layers (GDN vs. Residual Blocks) on the Kodak dataset. BD-Rate is computed against the GDN baseline. **Enc./Dec.** denotes inference time. *RBs* stands for residual blocks.

Layer	BD-Rate↓	Enc.(s)	Dec.(s)
GDN	0.00	**1.932**	**1.618**
RBs w/o Split-Atten	-7.22	2.371	1.671
RBs w/ Split-Atten	**-9.38**	3.443	1.698

D. Role of Split Attention vs. GDN

Our final ablation study evaluates the non-linear transformation block used within the residual CNN branch of our BACF module. In our design (Fig. 2), we diverged from the Generalized Divisive Normalization (GDN) [19] commonly used in learned compression, opting instead for residual bottleneck blocks (RBs) augmented with a Split-Attention (Split-Atten) mechanism [15]. To quantify the impact of this choice, we compared three models: (i) GDN: The traditional GDN layer is used as the non-linear unit. (ii) RBs w/o Split-Atten: Residual blocks are used, but without the split-attention module. (iii) RBs w/ Split-Atten: Our proposed design. For this test, models were configured with a 3-layer stack to maximize the observable effect of the non-linearity.

The results, presented in Table I, are illuminating. We use Bjontegaard Distortion-Rate (BD-Rate) [20] as the quantitative metric, with the GDN model serving as the anchor (0.00). Replacing GDN with standard residual blocks (RBs w/o Split-Atten) yields a significant BD-Rate improvement of -7.22%. However, augmenting these residual blocks with our chosen Split-Attention mechanism (RBs w/ Split-Atten) achieves a remarkable -9.38% reduction. This demonstrates that the

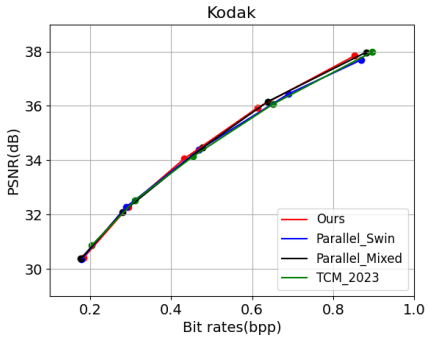

Fig. 7: Rate-distortion performance comparison on the Kodak dataset for different connection strategies. "Ours" represents the adopted Series connection.

split-attention module is a critical driver of performance in our CNN branch. While this design incurs a higher inference latency compared to the highly optimized GDN layer, the substantial bitrate saving at equivalent PSNR quality strongly justifies its inclusion for maximizing compression efficiency.

E. Connection Strategy: Series vs. Parallel

Finally, we analyze the connection strategy used to combine the BACF-Max and BACF-Swin blocks within each transformation layer. Our adopted framework (Fig. 1) utilizes a series (cascaded) connection, where the output of the BACF-Max block is sequentially fed into the BACF-Swin block. We compare this against an alternative parallel configuration, where the input feature map is fed to both the BACF-Max and BACF-Swin blocks simultaneously, and their respective outputs are fused via element-wise addition.

As shown in Fig. 7, the resulting R-D performance curves for the Series (labeled "Ours") and Parallel (labeled "Parallel, Swin" and "Parallel, Mixed") configurations are nearly identical. This observation suggests that the choice of connection topology for the internal transformer blocks does not yield a significant difference in compression efficiency. The performance gains primarily originate from the fusion design of the BACF block itself, rather than the specific method of stacking the internal attention modules. Given this negligible difference, we adopted the more straightforward Series connection for our final architecture.

IV. Experimental Evaluation

A. Implementation Details

To evaluate its performance, our optimized 2-layer BACF-Net was implemented using the CompressAI platform [21]. Following established practices for SOTA comparisons [5], [9], we trained our models on a large-scale dataset of 300,000 images randomly selected from the ImageNet training set [22]. During training, images were randomly cropped to 256×256 patches. We optimized the network using the Adam optimizer [23] with a batch size of 8.

We evaluate the R-D performance of our optimized model on two widely used benchmark datasets: (i) CLIC, specifically

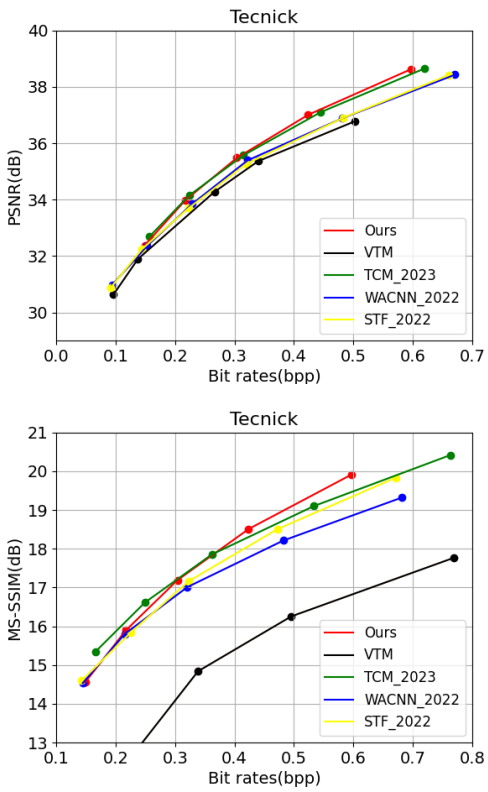

Fig. 8: Rate-distortion performance comparison on the Tecnick dataset. (Top) PSNR (dB) vs. bpp. (Bottom) MS-SSIM (dB) vs. bpp. Our optimized 2-layer BACF-Net demonstrates competitive performance.

the professional validation set (pro val), which consists of 41 high-quality images at 2K resolution; and (ii) Tecnick, which includes 100 high-resolution images (1200×1200). For clearer visual comparison on R-D plots, MS-SSIM values are converted to decibels (dB) using the formula $-10\log_{10}(1 - \text{MS-SSIM})$.

B. Performance Comparison

We benchmark the R-D performance of our optimized 2-layer BACF-Net against a comprehensive set of recent methods. Our comparisons include the traditional VVC (VTM-23) codec and leading learned compression frameworks, such as Ballé'18 [6], Cheng'20 [4], WACNN'22 [8], ELIC'22 [5], LIC_TCM'23 [9], and SCH'24 [24]. The results for these competing methods are sourced from their original publications or pre-trained models to ensure a fair comparison.

The Rate-Distortion (R-D) performance of our optimized 2-layer model is presented in Fig. 8 (Tecnick) and Fig. 9 (CLIC). Across all benchmark datasets, which feature diverse image resolutions and content, our optimized BACF-Net demonstrates competitive performance. The results highlight the robustness and effectiveness of our hybrid approach. The strategic combination of advanced residual CNN blocks, split attention, and mixed-transformer variants enables our model

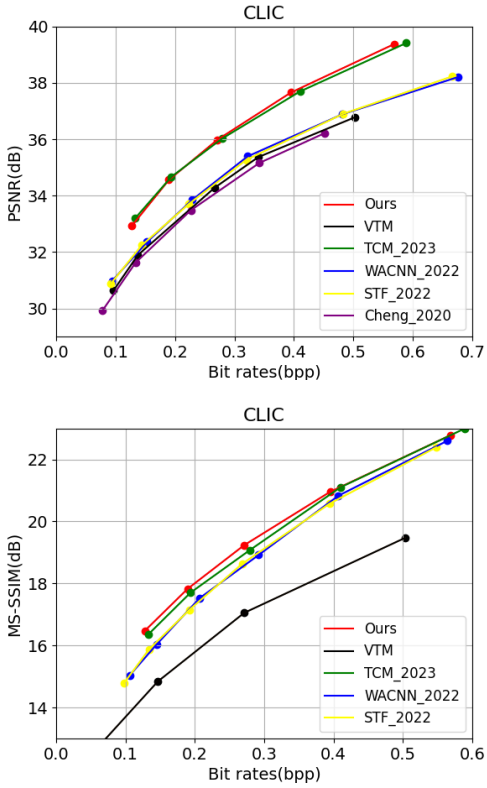

Fig. 9: Rate-distortion performance comparison on the CLIC Professional Validation dataset. (Top) PSNR (dB) vs. bpp. (Bottom) MS-SSIM (dB) vs. bpp.

to effectively balance the extraction of fine-grained local details with high-level semantic information. This allows our optimized framework to achieve a superior trade-off in both PSNR and MS-SSIM metrics, validating the efficacy of our architectural analysis and subsequent optimization.

V. Conclusion

In this paper, we presented a comprehensive architectural analysis and optimization of BACF-Net, a hybrid CNN-Transformer framework for learned image compression. Moving beyond the baseline performance established in our preliminary work, we provided a rigorous dissection of the key components driving the system's efficiency. This work confirms that a "dissect-then-optimize" approach is crucial for maximizing the potential of complex hybrid architectures.

References

[1] Gregory K Wallace. The jpeg still picture compression standard. *Communications of the ACM*, 34(4):30–44, 1991.

[2] Vivienne Sze, Madhukar Budagavi, and Gary J Sullivan. High efficiency video coding (hevc). In *Integrated circuit and systems, algorithms and architectures*, volume 39, page 40. Springer, 2014.

[3] Joint Video Experts Team. Vvc official test model vtm, 2021.

[4] Zhengxue Cheng, Heming Sun, Masaru Takeuchi, and Jiro Katto. Learned image compression with discretized gaussian mixture likelihoods and attention modules. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7939–7948, 2020.

[5] Dailan He, Ziming Yang, Weikun Peng, Rui Ma, Hongwei Qin, and Yan Wang. Elic: Efficient learned image compression with unevenly grouped space-channel contextual adaptive coding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5718–5727, 2022.

[6] Johannes Ballé, David Minnen, Saurabh Singh, Sung Jin Hwang, and Nick Johnston. Variational image compression with a scale hyperprior. In *International Conference on Learning Representations*, 2018.

[7] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

[8] Renjie Zou, Chunfeng Song, and Zhaoxiang Zhang. The devil is in the details: Window-based attention for image compression. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 17492–17501, 2022.

[9] Jinming Liu, Heming Sun, and Jiro Katto. Learned image compression with mixed transformer-cnn architectures. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14388–14397, 2023.

[10] Chen-Lin Chang and Hsu-Feng Hsiao. Bacf-net: An attention-convolution fusion architecture for learned image compression. In *2025 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 1–5, 2025.

[11] Zhengzhong Tu, Hossein Talebi, Han Zhang, Feng Yang, Peyman Milanfar, Alan Bovik, and Yinxiao Li. Maxvit: Multi-axis vision transformer. In *European conference on computer vision*, pages 459–479. Springer, 2022.

[12] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021.

[13] David Minnen, Johannes Ballé, and George D Toderici. Joint autoregressive and hierarchical priors for learned image compression. *Advances in neural information processing systems*, 31, 2018.

[14] David Minnen and Saurabh Singh. Channel-wise autoregressive entropy models for learned image compression. In *2020 IEEE International Conference on Image Processing (ICIP)*, pages 3339–3343. IEEE, 2020.

[15] Hang Zhang, Chongruo Wu, Zhongyue Zhang, Yi Zhu, Haibin Lin, Zhi Zhang, Yue Sun, Tong He, Jonas Mueller, R Manmatha, et al. Resnest: Split-attention networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2736–2746, 2022.

[16] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018.

[17] I Krasin, T Duerig, N Alldrin, V Ferrari, S Abu-El-Haija, A Kuznetsova, H Rom, J Uijlings, S Popov, A Veit, et al. Openimages: a public dataset for large-scale multi-label and multi-class image classification. dataset (2017), 2017.

[18] Zhengbo Luo, Zitang Sun, Weilian Zhou, Zizhang Wu, and Sei-ichiro Kamata. Rethinking resnets: improved stacking strategies with high-order schemes for image classification. *Complex & Intelligent Systems*, 8(4):3395–3407, 2022.

[19] Johannes Ballé, Valero Laparra, and Eero P Simoncelli. End-to-end optimization of nonlinear transform codes for perceptual quality. In *2016 Picture Coding Symposium (PCS)*, pages 1–5. IEEE, 2016.

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

[21] Jean Bégaint, Fabien Racapé, Simon Feltman, and Akshay Pushparaja. Compressai: a pytorch library and evaluation platform for end-to-end compression research. *arXiv preprint arXiv:2011.03029*, 2020.

[22] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[23] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[24] H. Xu, B. Hai, Y. Tang, and Z. He. Window-based channel attention for wavelet-enhanced learned image compression. In *Proceedings of the Asian Conference on Computer Vision*, pages 4334–4351, 2024.