This article has been accepted for publication in IEEE Transactions on Circuits and Systems for Video Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2026.3742841



# Dual-Branch Learned Image Compression with Conditional Entropy Modeling and Deformable Convolution-based Attention

Haisheng Fu, Feng Ding, Jie Liang*, Zhenman Fang, Jingning Han

***Abstract*—Learned image compression benefits from representations that jointly preserve large-scale structures and local details while remaining amenable to accurate entropy modeling. We present a dual-branch asymmetric framework in which two analysis paths use different receptive-field biases to form complementary structural and detail-oriented latent representations, followed by a shared synthesis transform. The structural latent is decoded first and serves as both a value prior and a distributional condition for the subsequent coding of the detail latent. A deformable convolution v2-based attention module is incorporated into the synthesis transform, the hyperprior analysis transform, and the entropy-parameter network to provide content-adaptive aggregation of cross-branch features. We further introduce a prior-utilization (PU) regularizer that explicitly encourages the conditional entropy model to obtain a rate advantage from the correctly matched decoded structural prior, while adding no auxiliary entropy path at inference. The final configuration also uses empirically selected implementation refinements for transform-depth allocation, fine-latent channel grouping, and higher-resolution fine-tuning; these refinements are evaluated separately and are not claimed as standalone methodological contributions. On Kodak, Tecnick, and CLIC Professional Validation, the final model achieves BD-Rate reductions of $-20.23\%$, $-23.58\%$, and $-23.46\%$ relative to the common H.266/VVC anchor used in our evaluation, respectively. Compared with several recent high-complexity LIC models, the proposed method provides a competitive tradeoff among compression efficiency, decoding latency, and computational complexity.**

***Index Terms*—Learned image compression, dual-branch representation, conditional entropy modeling, deformable convolution, rate–distortion optimization.**

## I. Introduction

IMAGE compression is fundamental to the storage and transmission of visual information. Conventional standards, including JPEG [1], JPEG2000 [2], HEVC [3], and VVC [4], remove statistical redundancy through engineered prediction, transform, quantization, and entropy-coding tools. Learned image compression (LIC) instead optimizes nonlinear transforms and probability models jointly under a rate–distortion objective [5]–[7].

Haisheng Fu, Feng Ding, Jie Liang, and Zhenman Fang are with the School of Engineering Science, Simon Fraser University, Canada (e-mails: hfa23@sfu.ca, feng_ding@sfu.ca, jiel@sfu.ca, zhenman@sfu.ca). *Corresponding author: Jie Liang.

Jingning Han is with Google Inc. (e-mail: jingning@google.com).

This work was supported by the Natural Sciences and Engineering Research Council of Canada (RGPIN-2020-04525), Google Chrome University Research Program, NSERC Discovery Grant RGPIN-2019-04613, DGECR-2019-00120, Alliance Grant ALLRP-552042-2020; CFI John R. Evans Leaders Fund; and MITACS Elevate Postdoc grant.

Recent LIC systems improve either representation learning or entropy modeling. Residual and attention-based transforms [8], [9], Transformer–CNN hybrids [10], and channel/spatial context models [11]–[13] have progressively improved coding efficiency. More recent models further explore linear attention [14], state-space modeling [15], and hierarchical progressive context [16]. These advances suggest that strong coding performance depends not only on network capacity, but also on how latent information is organized and what side information is available when each component is entropy-coded.

Most learned codecs employ a single analysis pathway and allow the network to discover representations at multiple spatial scales implicitly. Such a design is highly expressive, but it does not explicitly separate features with different receptive-field requirements. Our starting point is therefore not the claim that a single branch cannot learn multi-scale features; rather, we investigate whether an explicit coarse-to-fine decomposition can provide a useful inductive bias for conditional coding. The proposed encoder contains a $5 \times 5$ branch with a broader local receptive field and a $3 \times 3$ branch with a more localized receptive field. Their outputs are referred to as structural and detail-oriented latents, respectively. These terms describe their empirically observed roles rather than a strict frequency decomposition. Compared with the closely related frequency-decomposition model FDNet [17], our two branches are differentiated by their receptive-field characteristics and are not explicitly constrained to represent disjoint or orthogonal frequency bands.

The key coding dependency is imposed at the entropy-model level. The structural latent is decoded first, and the information available from this decoded stream is then used when estimating the probability distribution of the detail latent. This ordering makes the cross-branch information flow decoder-realizable: every condition used to code a detail group is available identically at the encoder and decoder. We additionally employ deformable convolution v2-based attention (DCNv2-Atten) to aggregate cross-branch features with content-adaptive sampling in the synthesis and probability-estimation paths.

A practical failure mode of conditional coding arises when an expressive entropy model relies predominantly on the hyperprior and previously decoded detail groups while making limited use of the decoded structural condition. To address this under-utilization, we introduce a rate-based prior-utilization regularizer that compares the conditional negative

Authorized licensed use limited to: SIMON FRASER UNIVERSITY. Downloaded on November 19,2026 at 23:45:16 UTC from IEEE Xplore. Restrictions apply.
© 2026 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Circuits and Systems for Video Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2026.3742841



log-likelihood obtained with the correctly matched structural prior against a content-mismatched reference. The objective is used only during training and leaves the deployed entropy factorization unchanged.

Our main contributions are summarized as follows:

- Building on the dual-branch representation introduced in our conference work, we establish a decoder-realizable coarse-to-fine entropy-coding dependency: the structural latent is decoded first and supplies both reconstructed values and learned distributional information to the conditional probability model of the detail latent. The methodological extension therefore lies in the ordered cross-branch representation–entropy dependency rather than in the existence of two analysis pathways alone.
- We adapt DCNv2-based attention as a content-adaptive cross-branch aggregation mechanism and deploy it at three codec locations: the synthesis transform, the hyperprior analysis transform, and the entropy-parameter network. Importantly, the conditional information used for entropy decoding is restricted to decoder-available context.
- We introduce a prior-utilization regularizer that explicitly promotes a coding advantage for the correctly matched decoded structural prior. The regularizer operates on conditional negative log-likelihoods, is training-only, and is directly evaluated through fine-stream and total-rate measurements.

Several performance-oriented refinements are used in the final configuration, including a nonuniform stage-wise TCM depth schedule, finer channel grouping of the detail latent, and higher-resolution second-stage fine-tuning. We deliberately do not present these choices as independent contributions. Instead, they are documented in Sec. III-D and isolated in Sec. IV-E so that the final rate–distortion improvement is experimentally traceable. The tradeoff among decoding time, model complexity, and BD-Rate reductions relative to H.266/VVC is further illustrated in Fig. 1 through a visual comparison on the Kodak dataset.

This work extends our previous conference paper [18]. Relative to that version, the journal model replaces the $1\times1/3\times3$ analysis design with $5\times5/3\times3$ branches, reverses the coding order so that the structural latent is decoded first, transfers both value and distribution priors across branches, incorporates DCNv2-Atten into reconstruction and probability estimation, and uses 160-channel structural and detail latents. It further introduces the PU regularizer and a substantially expanded experimental analysis that separates the core method from performance-oriented implementation refinements.

## II. Related Work

*a) Representation and transform design.:* Autoencoder-based LIC learns nonlinear analysis and synthesis transforms under a differentiable rate–distortion objective [5], [6], [19]. Residual blocks, attention mechanisms, invertible networks, and Transformer-based transforms have all been used to increase modeling capacity [8], [10], [20], [21]. A closely related study, Asymmetric LIC [9], primarily targets encoder–decoder complexity asymmetry, importance scaling, and post-quantization filtering; our central asymmetry instead lies in the ordered entropy-coding dependency between separately generated structural and detail representations. Wavelet-like and frequency-aware approaches provide alternative inductive biases for separating image components [22], [23].

*b) Conditional entropy modeling.:* Hyperpriors transmit side information for latent-distribution estimation [6]; autoregressive models additionally exploit decoded context [7]. Channel-wise autoregression [11], checkerboard context [24], and multi-reference entropy models [12], [25] improve the balance between probability accuracy and serial dependency. Recent models further exploit group-wise mixing [13], linear attention [14], and hierarchical progressive context [16]. Within closely related entropy-modeling work, GroupedMixer [13] strengthens interactions among groups of a single latent representation, whereas our probability model transfers decoder-available information across two separately generated representation branches under a fixed structural-to-detail decoding order.

Causal Context Adjustment Loss (CCA) [26] demonstrates that auxiliary objectives can improve utilization of causal context. Our PU regularizer is related in spirit but targets a different dependency: the condition being tested is a separately coded structural latent, and the reference is formed by deliberately breaking its content correspondence with the detail target. We do not claim that context-oriented auxiliary losses are new in general.

*c) Adaptive feature aggregation.:* Attention has been widely used in learned compression to reweight spatial and channel features [8], [27], [28]. DCNv2 [29] provides content-adaptive sampling through learned offsets and modulation. In this work, DCNv2 is not introduced as a new operator; it is adapted to aggregate features produced by two branches with different receptive-field biases and is inserted only where the corresponding context is available in the codec.

## III. The Proposed Image Compression Framework

### A. Dual-Branch Representation and Adaptive Aggregation

The complete codec is illustrated in Fig. 2. The input image $\boldsymbol{x} \in \mathbb{R}^{H\times W\times 3}$ is processed by two analysis transforms, $g_{a,CS}$ and $g_{a,FS}$, that generate latent representations $y_{CS}$ and $y_{FS}$. The structural branch uses $5\times5$ convolutions to provide a broader local receptive-field bias, whereas the detail branch uses $3\times3$ convolutions. Both branches contain TCM processing stages [10] and four downsampling operations. A shared synthesis transform $g_s$ reconstructs $\hat{\boldsymbol{x}}$ from the decoded latents.

The two latents have the same nominal spatial resolution after the analysis transforms. Consequently, the term "cross-branch alignment" in this paper refers to adaptive aggregation of feature responses generated with different receptive-field biases, rather than to registration between different geometric coordinate systems. This distinction is important when interpreting the role of DCNv2-Atten.

Authorized licensed use limited to: SIMON FRASER UNIVERSITY. Downloaded on November 19,2026 at 23:45:16 UTC from IEEE Xplore. Restrictions apply.
© 2026 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Circuits and Systems for Video Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2026.3742841



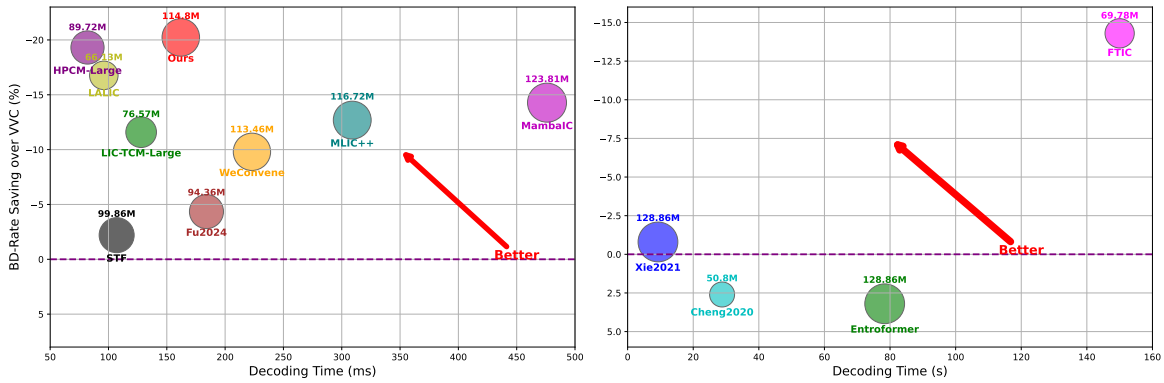



Fig. 1. The decoding time, model complexity, and BD-Rate reductions relative to H.266/VVC of various learned image compression methods evaluated on the Kodak dataset. The circle size indicates model parameters, which is also displayed. Note the different x-axis units in the two subfigures: milliseconds (ms) in the left and seconds (s) in the right. Our approach demonstrates a competitive trade-off among these three performance metrics.

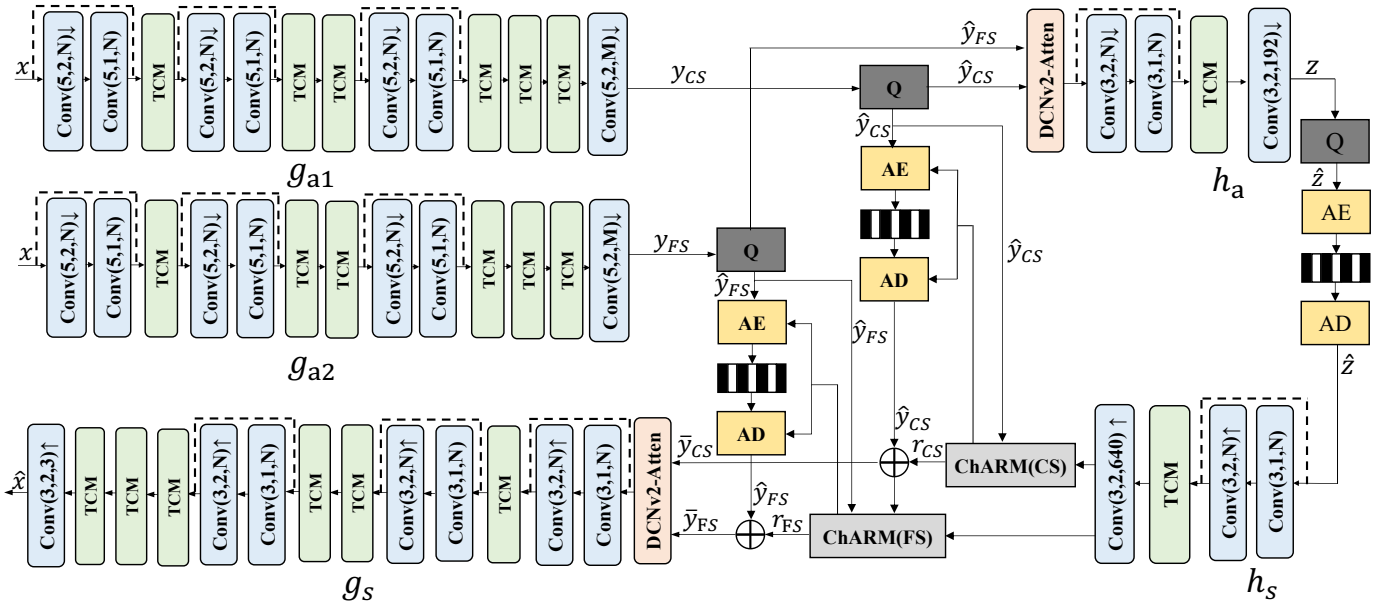



Fig. 2. Overall architecture of the proposed dual-branch learned image codec. The two analysis branches use different receptive-field biases and produce structural ($y_{CS}$) and detail-oriented ($y_{FS}$) latents on the same nominal spatial grid. The hyperlatent is decoded first, followed by $y_{CS}$ and then $y_{FS}$. ChARM denotes the channel-wise autoregressive entropy model; TCM denotes the Transformer–CNN Mixture block [10]; AE/AD denote arithmetic encoding/decoding. Exact stage-wise TCM depths and channel-group counts used by the final configuration are specified in Secs. III-B and III-D.

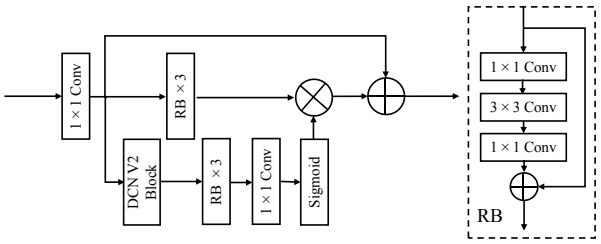



Fig. 3. DCNv2-Atten module used for content-adaptive feature aggregation.

Let $F$ denote an input feature map. A modulated deformable convolution computes

$$F'(p_0) = \sum_{k=1}^{K} w_k \, F(p_0 + p_k + \Delta p_k) \, \Delta m_k, \qquad (1)$$

where $p_k$ denotes a regular sampling offset, $\Delta p_k$ is a learned content-dependent displacement, and $\Delta m_k \in [0, 1]$ is a modulation scalar. DCNv2-Atten retains the channel-squeeze and aggregation structure of SWAtten [10], while replacing its fixed-window Transformer component with the modulated deformable operator. The module is inserted in the synthesis transform, the hyperprior analysis transform, and the entropy-parameter networks. The purpose is to improve content-adaptive aggregation; no novelty is claimed for Eq. (1) itself.

### B. Cross-Branch Conditional Entropy Model

The coding order is central to the proposed formulation. The hyperlatent $\hat{z}$ is decoded first. The structural latent $\hat{y}_{CS}$ is then decoded using the hyperprior, and only after $\hat{y}_{CS}$ is available does the codec decode the detail latent $\hat{y}_{FS}$. Therefore the

Authorized licensed use limited to: SIMON FRASER UNIVERSITY. Downloaded on November 19,2026 at 23:45:16 UTC from IEEE Xplore. Restrictions apply.
© 2026 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Circuits and Systems for Video Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2026.3742841



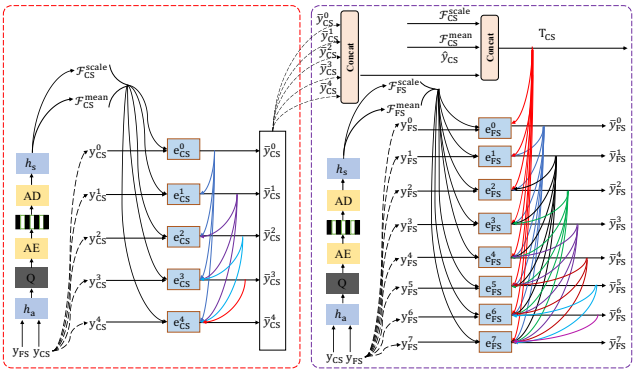



Fig. 4. Cross-branch channel-wise autoregressive entropy model. The structural latent is decoded before the detail latent. The displayed group count is schematic; the final model uses five structural slices and eight detail groups, as specified in Sec. III-B.

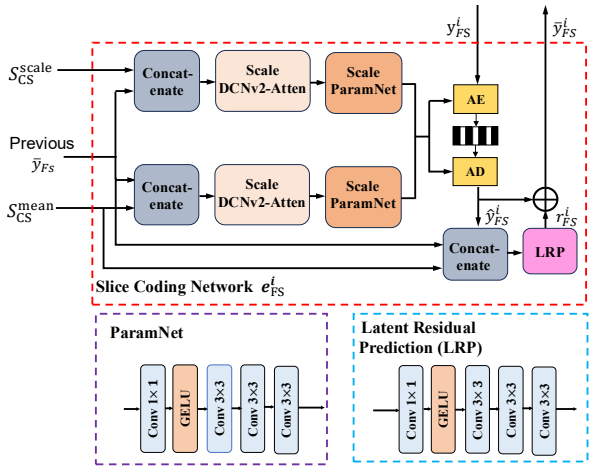



Fig. 5. Detailed entropy-parameter network used to code one group of the detail latent.

implemented probability factorization is

$$p(\hat{z}, \hat{y}_{CS}, \hat{y}_{FS}) = p(\hat{z})\, p(\hat{y}_{CS} \mid \hat{z})\, p(\hat{y}_{FS} \mid \hat{y}_{CS}, \hat{z}). \quad (2)$$

For the same pair of random variables, the conditional-entropy chain rule gives

$$H(Y_{CS}, Y_{FS} \mid Z) = H(Y_{CS} \mid Z) + H(Y_{FS} \mid Y_{CS}, Z). \quad (3)$$

Equation (3) motivates the ordered code but does not, by itself, prove that one learned representation has lower entropy than a differently trained single-stream representation. The practical benefit depends on whether $Y_{CS}$ contains predictive information that the learned probability model can exploit when coding $Y_{FS}$. This observation motivates the PU objective in Sec. III-C.

Both latent tensors contain 160 channels in the final model. The structural stream is partitioned into five 32-channel slices, $\{y^i_{CS}\}_{i=1}^{5}$. Each slice is decoded from hyperprior side information and the previously decoded structural slices. The reconstructed structural representation and its probability-model features then become available to the detail coder as value and distributional conditions.

The detail latent is partitioned more finely into eight 20-channel groups, $\{y^g_{FS}\}_{g=1}^{8}$. For group $g$, the entropy-parameter network conditions on $\hat{z}$, the decoded structural representation, and previously decoded detail groups $\hat{y}^{<g}_{FS}$. The narrower group width increases the granularity at which newly decoded detail context becomes available. The eight-group choice is an empirically selected implementation setting and is evaluated against five- and ten-group controls in Sec. IV-E.

The entropy model also predicts latent residuals that refine the quantized representations before reconstruction, following the latent-residual prediction strategy used in ChARM-based models [11]. All inputs to the entropy-parameter network for the current group are decoder-available; the uncoded target group itself is never used to predict its own probability parameters.

### C. Prior-Utilization Regularization

The standard rate–distortion objective is

$$\mathcal{L}_{RD} = \lambda D(\boldsymbol{x}, \hat{\boldsymbol{x}}) + R_z + R_{CS} + R_{FS}, \quad (4)$$

where

$$R_z = \mathbb{E}[-\log_2 p(\hat{z})], \quad (5)$$

$$R_{CS} = \mathbb{E}[-\log_2 p(\hat{y}_{CS} \mid \hat{z})], \quad (6)$$

$$R_{FS} = \sum_{g=1}^{G_{FS}} \mathbb{E}[-\log_2 p_\theta(\hat{y}^g_{FS} \mid \hat{y}^{<g}_{FS}, \hat{y}_{CS}, \hat{z})]. \quad (7)$$

An expressive conditional model can in principle underutilize $\hat{y}_{CS}$ if the hyperprior and previously decoded detail groups already provide strong predictors. To explicitly encourage useful cross-branch conditioning, we compare the per-symbol negative log-likelihood under the matched structural condition with a mismatched reference. Let $\pi$ be a no-fixed-point derangement of mini-batch indices. For detail group $g$,

$$\rho^+_g = -\frac{1}{|\Omega_g|} \log_2 p_\theta(\hat{y}^g_{FS} \mid \hat{y}^{<g}_{FS}, \hat{y}_{CS}, \hat{z}), \quad (8)$$

$$\rho^-_g = -\frac{1}{|\Omega_g|} \log_2 p_\theta(\hat{y}^g_{FS} \mid \hat{y}^{<g}_{FS}, \pi(\hat{y}_{CS}), \hat{z}), \quad (9)$$

where $|\Omega_g|$ is the number of symbols in group $g$. The derangement preserves the set of structural samples within the mini-batch while deliberately breaking their content correspondence with the target detail sample. It is used only as a training reference and is not interpreted as a valid deployed coding distribution. We define

$$\mathcal{L}_{PU} = \frac{1}{G_{FS}} \sum_{g=1}^{G_{FS}} \left[ m + \rho^+_g - \mathrm{sg}(\rho^-_g) \right]_+, \quad (10)$$

where $m$ is a target margin, $[u]_+ = \max(u, 0)$, and $\mathrm{sg}(\cdot)$ denotes stop-gradient. When the margin is violated, the active hinge increases the gradient pressure to reduce the matched conditional rate. Stop-gradient prevents a direct optimization path through the mismatched reference in that update; it does not imply that the negative reference is invariant to later updates of shared model parameters.

The final training objective is

$$\mathcal{L} = \mathcal{L}_{RD} + \eta \mathcal{L}_{PU}. \quad (11)$$

Authorized licensed use limited to: SIMON FRASER UNIVERSITY. Downloaded on November 19,2026 at 23:45:16 UTC from IEEE Xplore. Restrictions apply.
© 2026 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Circuits and Systems for Video Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2026.3742841



PU is therefore an auxiliary optimization term rather than a second entropy model. The deployed codec follows the single factorization in Eq. (2) and incurs no additional mismatched-prior pass.

### D. Implementation and Training Details

The training images are drawn from the CLIC training set and LIU4K [30]. We use random rotation, scaling, and cropping and train separate models for MSE and MS-SSIM optimization. For MSE, $\lambda \in \{0.0016, 0.0032, 0.0075, 0.015, 0.03, 0.045, 0.06\}$; for MS-SSIM, $\lambda \in \{12, 40, 80, 120\}$. The base filter setting is $N = 128$, while both final latent tensors contain 160 channels.

Let $\boldsymbol{d}_a$ and $\boldsymbol{d}_s$ denote the TCM depth vectors of the three analysis and synthesis processing stages, respectively, ordered from high to low spatial resolution for the analysis transform and from low to high spatial resolution for the synthesis transform. The final implementation uses $\boldsymbol{d}_a = (1, 2, 3)$ and $\boldsymbol{d}_s = (3, 2, 1)$. This redistributes contextual modeling toward smaller feature maps while preserving six TCM blocks per path relative to the equal-depth control in Sec. IV-E. The depth vectors are implementation parameters whose effect is quantified experimentally, not standalone methodological contributions.

Training uses two stages. Stage I optimizes $384 \times 384$ crops for $10^6$ iterations with Adam, batch size 8, and learning rate $10^{-4}$, using only $\mathcal{L}_{RD}$. Stage II continues training for $5 \times 10^5$ iterations on $512 \times 512$ crops with batch size 4 and learning rate $10^{-5}$. PU is activated only in Stage II with $\eta = 0.05$ and $m = 0.02$ bits per latent symbol. The larger-crop schedule and exact group count are treated as performance-oriented refinements; their measured effects are reported rather than presented as methodological novelty.

## IV. Experimental Results

### A. Experimental Setup

We evaluate rate–distortion performance using PSNR and MS-SSIM on Kodak [31], Tecnick [32], and CLIC Professional Validation [33]. Kodak contains 24 images at $768 \times 512$ resolution, Tecnick contains 100 images at $1200 \times 1200$, and CLIC Professional Validation contains 41 images with resolutions up to 2K. The compared learned codecs include Cheng2020 [8], ELIC [12], LIC-TCM-Large [10], FTIC [23], MLIC++ [25], MambaIC [15], LALIC [14], CCA [26], and HPCM-Large [16], together with the additional methods listed in Table I.

All BD-Rate values in Table I use a common H.266/VVC anchor and the same evaluation protocol as the preceding comparison. Competitor measurements are retained from that matched evaluation rather than re-estimated under a different setup. For the proposed method, the R-D curves and the reported BD-Rate values are computed from the same final-model operating points and corresponding encoded bitstreams. Encoding/decoding times for learned codecs were measured with released pretrained models on an NVIDIA RTX 4090 GPU, while VVC was evaluated on a 2.9 GHz Intel Xeon Gold 6226R CPU. Timings correspond to actual bitstream generation and decoding rather than neural-network forward time alone.

### B. Rate-Distortion Performances

Fig. 6 shows the average rate-distortion curves of different methods on the Kodak dataset under both PSNR and MS-SSIM optimization. For PSNR, our proposed method achieves the best overall rate-distortion performance among the compared learned image compression methods. At low bit rates, its performance is very close to HPCM-Large [16], with the two curves nearly overlapping. As the bit rate increases, our method gradually exhibits a small but consistent advantage over HPCM-Large, while also outperforming other recent strong baselines, including LALIC [14], MambaIC [15], CCA [26], and LIC-TCM-Large [10]. This trend is consistent with the BD-Rate results in Table I, where our method achieves a slightly lower BD-Rate than HPCM-Large on Kodak.

A similar trend can be observed under the MS-SSIM metric. At the low-rate operating points, our method performs almost identically to HPCM-Large, indicating that the proposed model preserves its strong low-rate coding efficiency. At medium and high bit rates, our method gradually achieves a slight advantage over HPCM-Large, while maintaining lower bit rates for comparable MS-SSIM quality. It also consistently outperforms LALIC [14], CCA [26], and the remaining compared methods over most of the evaluated range. These results demonstrate that the proposed method provides consistent improvements under both PSNR- and perceptual-quality-oriented optimization, while the performance gain over HPCM-Large remains moderate rather than being concentrated at a particular operating point.

We further evaluate the rate-distortion performance on the Tecnick and CLIC Professional Validation datasets, as shown in Fig. 7. On the Tecnick dataset, our method again closely follows HPCM-Large at low bit rates and gradually achieves a small advantage as the bit rate increases. It also consistently outperforms LALIC [14], CCA [26], LIC-TCM-Large [10], and the other compared learned compression methods over the evaluated range. The corresponding BD-Rate results further confirm this improvement, with our method achieving a slightly better overall coding efficiency than HPCM-Large.

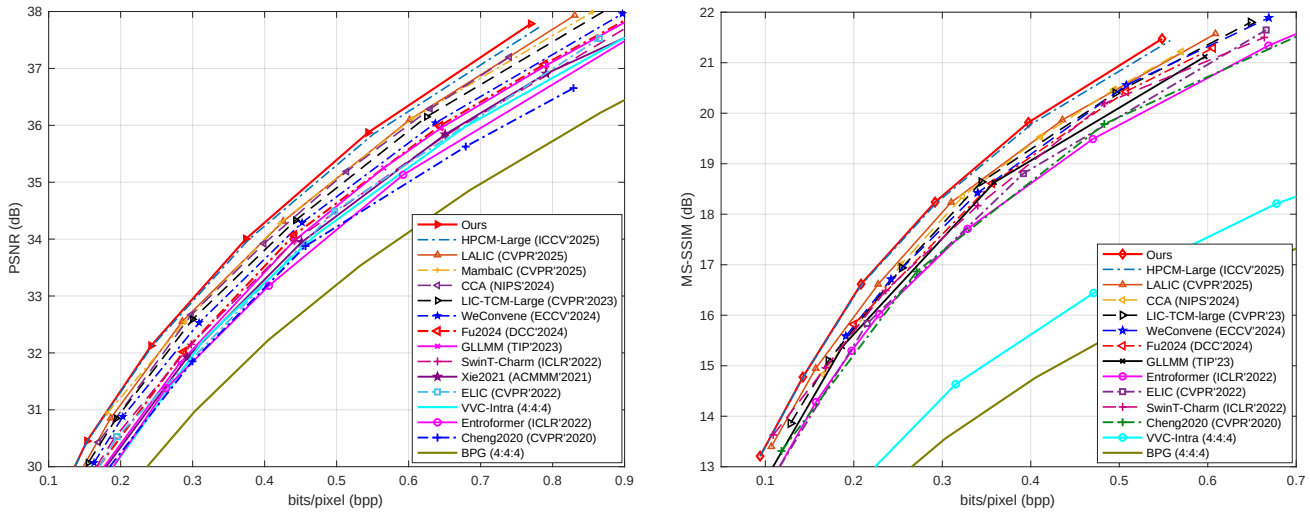
On the CLIC Professional Validation dataset, the proposed method exhibits a similar rate-distortion behavior. Its low-rate performance remains very close to that of HPCM-Large, whereas a modest advantage gradually emerges toward the medium- and high-rate regions. Our method also achieves consistently better performance than LALIC [14], CCA [26], LIC-TCM-Large [10], and the remaining learned baselines across most of the evaluated bitrate range. This observation is also consistent with the BD-Rate comparison reported in Table I.

Overall, the results on Kodak, Tecnick, and CLIC Professional Validation demonstrate that the proposed method achieves consistently strong rate-distortion performance across datasets with different image characteristics and resolutions. Compared with the previous best-performing HPCM-Large

Authorized licensed use limited to: SIMON FRASER UNIVERSITY. Downloaded on November 19,2026 at 23:45:16 UTC from IEEE Xplore. Restrictions apply.
© 2026 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

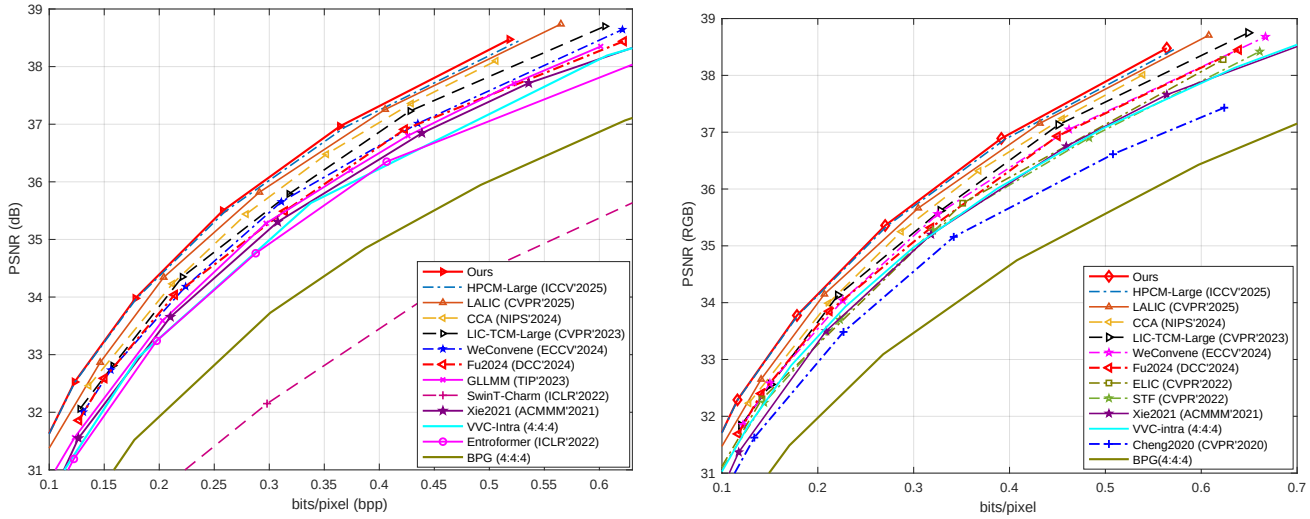
This article has been accepted for publication in IEEE Transactions on Circuits and Systems for Video Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2026.3742841






(a) (b)

Fig. 6. Average rate–distortion performance on Kodak under (a) PSNR and (b) MS-SSIM optimization.




(a) (b)

Fig. 7. Average PSNR performance on (a) Tecnick and (b) CLIC Professional Validation.

model, the proposed method maintains nearly identical performance in the low-rate region while providing small but consistent gains at medium and high bit rates. The consistent behavior across both PSNR and MS-SSIM metrics further indicates that the proposed improvements are not limited to a specific distortion objective or dataset.

### *C. Tradeoff between Compression Efficiency and Computational Complexity*

Table I presents a comprehensive comparison of several recent LIC methods in terms of encoding time, decoding time, BD-Rate reduction over the VVC baseline [39], computational complexity (KMACs/pixel), and the number of model parameters in millions (M). All learned codecs were evaluated on an NVIDIA RTX 4090 GPU with 24 GB of memory, while H.266/VVC was evaluated on a 2.9 GHz Intel Xeon Gold 6226R CPU. The encoding and decoding time results are obtained using the pretrained models released by the original authors. Notably, the reported timings correspond to the actual encoding-to-bitstream and decoding-from-bitstream processes, instead of only the neural network inference time.

Several LIC methods, such as Cheng2020 [8] and FTIC [23], rely on serial spatial context adaptive entropy models to achieve strong coding performance. However, this design can significantly increase encoding and decoding time. By contrast, recent LIC methods, including ELIC [12], LIC-TCM-Large [10], and WeConvene [38], adopt GPU-friendly and parallelizable entropy models, which substantially reduce decoding latency.

Our method achieves BD-Rates of $-20.23\%$, $-23.58\%$, and $-23.46\%$ on the Kodak, Tecnick, and CLIC Professional Validation datasets, respectively, consistently achieving the lowest BD-Rate among the compared learned image compression

Authorized licensed use limited to: SIMON FRASER UNIVERSITY. Downloaded on November 19,2026 at 23:45:16 UTC from IEEE Xplore. Restrictions apply.
© 2026 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Circuits and Systems for Video Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2026.3742841



TABLE I
COMPREHENSIVE COMPARISON OF ENCODING TIME, DECODING TIME, BD-RATE RELATIVE TO THE COMMON VVC ANCHOR, MODEL PARAMETERS, AND COMPUTATIONAL COMPLEXITY. THE PROPOSED METHOD IS REPRESENTED ONLY BY THE FINAL REVISED MODEL; EARLIER CONFIGURATIONS ARE REPORTED EXCLUSIVELY IN THE ABLATION STUDIES. LOWER IS BETTER.

| Dataset | Method | Enc. (s) ↓ | Dec. (s) ↓ | BD-Rate ↓ | #Params (M) ↓ | KMACs/pixel ↓ |
|---|---|---|---|---|---|---|
| **Kodak** | VVC | 402.3 | 0.61 | 0.0 | – | – |
| | Cheng2020 [8] | 27.6 | 28.8 | 2.6% | 50.80 | 1027.22 |
| | Hu2021 [34] | 32.7 | 77.8 | 11.1% | 84.60 | – |
| | He2021 [24] | 20.4 | 5.2 | 8.9% | 46.60 | – |
| | Xie2021 [21] | 4.10 | 9.25 | -0.8% | 128.86 | 1038.20 |
| | SwinT-Charm [35] | 0.27 | 0.18 | -3.9% | **32.34** | – |
| | ELIC [12] | 0.175 | 0.094 | -4.69% | 33.29 | 573.88 |
| | STF [36] | 0.106 | 0.107 | -2.2% | 99.86 | **511.17** |
| | Entroformer [37] | 1.50 | 78.37 | 3.2% | 128.86 | – |
| | GLLMM [20] | 420.6 | 423.8 | -3.1% | – | – |
| | Fu2024 [18] | 0.161 | 0.184 | -4.35% | 94.36 | 1092.54 |
| | LIC-TCM-Large [10] | 0.120 | 0.128 | -11.59% | 76.57 | 1823.58 |
| | WeConvene [38] | 0.222 | 0.223 | -9.8% | 113.46 | 2343.11 |
| | FTIC [23] | >150 | >150 | -14.38% | 69.78 | 1096.04 |
| | CCA [26] | 0.123 | 0.098 | -13.23% | 64.90 | 1250.84 |
| | LALIC [14] | 0.193 | 0.096 | -16.77% | 66.13 | 667.26 |
| | MLIC++ [25] | 0.241 | 0.309 | -12.69% | 116.72 | 1282.81 |
| | MambaIC [15] | 0.573 | 0.476 | -14.28% | 123.81 | 1685.08 |
| | HPCM-Large [16] | **0.062** | **0.082** | -19.31% | 89.72 | 1261.29 |
| | **Ours** | 0.188 | 0.162 | **-20.23%** | 114.80 | 1315.40 |
| **Tecnick** | VVC | 700.59 | 1.49 | 0.0 | – | – |
| | STF [36] | 0.431 | 0.472 | -2.6% | 99.90 | **511.17** |
| | ELIC [12] | 0.568 | 0.360 | -10.86% | **33.29** | 573.88 |
| | Fu2024 [18] | 0.477 | 0.469 | -3.1% | 94.36 | 1092.54 |
| | LIC-TCM-Large [10] | 0.357 | 0.374 | -12.80% | 76.57 | 1823.58 |
| | WeConvene [38] | 0.766 | 0.672 | -12.31% | 113.46 | 2343.11 |
| | FTIC [23] | >630 | >630 | -16.15% | 69.78 | 1096.04 |
| | CCA [26] | 0.383 | 0.225 | -15.75% | 64.90 | 1250.84 |
| | LALIC [14] | 0.617 | 0.307 | -17.62% | 66.13 | 667.26 |
| | HPCM-Large [16] | **0.181** | **0.174** | -22.85% | 89.72 | 1261.29 |
| | **Ours** | 0.605 | 0.457 | **-23.58%** | 114.80 | 1315.40 |
| **CLIC Professional Validation** | VVC | 949.58 | 1.98 | 0.0 | – | – |
| | STF [36] | 0.403 | 0.378 | 0.79% | 99.90 | **511.17** |
| | ELIC [12] | 0.515 | 0.258 | -3.43% | **33.29** | 573.88 |
| | Fu2024 [18] | 0.672 | 0.715 | -3.1% | 94.36 | 1092.54 |
| | LIC-TCM-Large [10] | 0.562 | 0.539 | -7.68% | 76.57 | 1823.58 |
| | WeConvene [38] | 1.207 | 1.044 | -5.58% | 113.46 | 2343.11 |
| | FTIC [23] | >10000 | >10000 | -9.63% | 69.78 | 1096.04 |
| | CCA [26] | 0.728 | 0.502 | -10.67% | 64.90 | 1250.84 |
| | LALIC [14] | 0.921 | 0.436 | -14.34% | 66.13 | 667.26 |
| | HPCM-Large [16] | **0.288** | **0.280** | -22.85% | 89.72 | 1261.29 |
| | **Ours** | 0.906 | 0.650 | **-23.46%** | 114.80 | 1315.40 |

methods on all three datasets. In particular, compared with the recent HPCM-Large model, which achieves $-19.31\%$, $-22.85\%$, and $-22.85\%$ on Kodak, Tecnick, and CLIC Professional Validation, respectively, our method further improves the BD-Rate by 0.92, 0.73, and 0.61 percentage points. These relatively moderate but consistent gains agree well with the rate-distortion curves, where our method remains close to HPCM-Large in the low-rate region and gradually exhibits a small advantage at medium and high bit rates. Our method also consistently surpasses other recent strong baselines, including LALIC [14], MLIC++ [25], MambaIC [15], and CCA [26], in terms of overall BD-Rate performance wherever corresponding results are available.

The improved compression efficiency is obtained with a moderate increase in computational cost compared with the most lightweight recent alternatives. Our model contains 114.80 M parameters and requires 1315.40 KMACs/pixel. For example, HPCM-Large uses 89.72 M parameters and

Authorized licensed use limited to: SIMON FRASER UNIVERSITY. Downloaded on November 19,2026 at 23:45:16 UTC from IEEE Xplore. Restrictions apply.
© 2026 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Circuits and Systems for Video Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2026.3742841



1261.29 KMACs/pixel, while LALIC requires only 66.13 M parameters and 667.26 KMACs/pixel. Therefore, the main advantage of the proposed model lies in its improved rate-distortion efficiency rather than achieving the lowest computational complexity. Nevertheless, the proposed architecture maintains practical GPU-based encoding and decoding times, requiring 0.188/0.162 s, 0.605/0.457 s, and 0.906/0.650 s for encoding/decoding on Kodak, Tecnick, and CLIC Professional Validation, respectively. These results demonstrate that the proposed method provides a favorable compression-performance tradeoff, as shown in Fig. 1, achieving the best BD-Rate among the compared methods while retaining practical GPU inference and entropy-coding latency.

Compared with our previous conference paper [18], the proposed method substantially improves compression efficiency on all three datasets. On Kodak, the BD-Rate improves from $-4.35\%$ to $-20.23\%$, while the decoding time decreases from 0.184 s to 0.162 s. On Tecnick, the BD-Rate improves from $-3.10\%$ to $-23.58\%$, and the decoding time decreases slightly from 0.469 s to 0.457 s. On CLIC Professional Validation, the BD-Rate improves from $-3.10\%$ to $-23.46\%$, while the decoding time decreases from 0.715 s to 0.650 s. The encoding time increases moderately from 0.161 s to 0.188 s on Kodak, from 0.477 s to 0.605 s on Tecnick, and from 0.672 s to 0.906 s on CLIC Professional Validation. The parameter count increases from 94.36 M to 114.80 M parameters, while the computational complexity increases from 1092.54 to 1315.40 KMACs/pixel.

Overall, compared with our previous conference model, our framework primarily targets substantially improved compression efficiency while preserving practical decoding speed, at the cost of a moderate increase in parameter count, computational complexity, and encoding time. More importantly, the improvements are consistent across Kodak, Tecnick, and CLIC Professional Validation, demonstrating that the proposed modifications generalize well across datasets with different image resolutions and content characteristics.

### *D. Component Ablations of the Base Dual-Branch Model*

We first analyze a base dual-branch model, denoted $B_0$, which represents the proposed architecture before the final performance-oriented refinements. $B_0$ uses the $5 \times 5/3 \times 3$ analysis branches, 160-channel structural and detail latents, five 32-channel slices in each latent, three TCM blocks in each analysis path and in the synthesis path, cross-branch prior integration, and DCNv2-Atten. The measurements in this subsection characterize this base model and its component variants; the final model is evaluated separately in Sec. IV-E. We first analyze a base dual-branch model, denoted $B_0$, which represents the proposed architecture before the final performance-oriented refinements. $B_0$ uses the $5 \times 5/3 \times 3$ analysis branches, 160-channel structural and detail latents, five 32-channel slices in each latent, three TCM blocks in each analysis path and in the synthesis path, cross-branch prior integration, and DCNv2-Atten. The measurements in this subsection characterize this base model and its component variants; the final model is evaluated separately in Sec. IV-E.

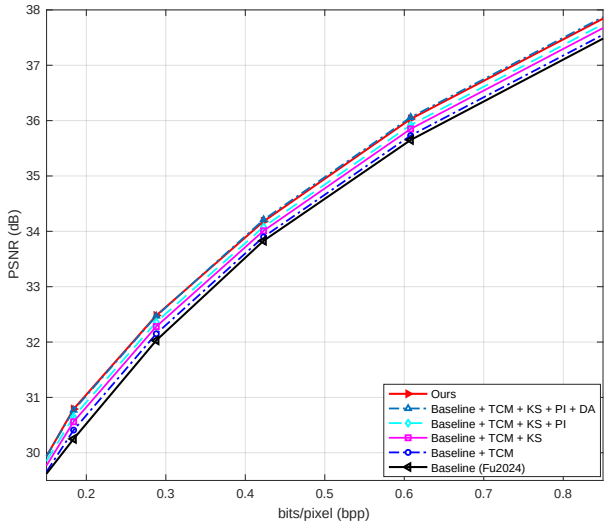



Fig. 8. R-D comparison of components leading to the base dual-branch model $B_0$.

TABLE II
COMPONENT ABLATION ON KODAK FOR THE BASE DUAL-BRANCH MODEL $B_0$.

| Configuration | Enc. | Dec. | BD-Rate | #Params (M) |
|---|---|---|---|---|
| Baseline [18] | **0.161s** | 0.184s | -4.35% | 94.36 |
| + TCM | 0.178s | 0.196s | -7.29% | 110.30 |
| + kernel redesign | 0.182s | 0.203s | -9.83% | 117.80 |
| + cross-branch prior integration | 0.185s | 0.204s | -11.54% | 120.00 |
| + DCNv2-Atten | 0.187s | 0.207s | **-13.72%** | 128.30 |
| Base model $B_0$ (160-channel latents) | 0.166s | **0.134s** | -12.91% | **91.86** |

TABLE III
LOCATION-SPECIFIC DCNV2-ATTEN ABLATION ON KODAK FOR THE BASE MODEL $B_0$.

| Configuration | BD-Rate | #Params (M) |
|---|---|---|
| Without DCNv2-Atten | -10.06% | 83.56 |
| Hyperprior analysis only | -10.57% | 85.80 |
| Entropy model only | -10.96% | 85.81 |
| Synthesis transform only | -11.63% | 86.17 |
| All three locations | **-12.91%** | 91.86 |

The sequence in Table II is descriptive rather than additive: later rows change more than one practical property of the codec. Kernel redesign improves the BD-Rate from -7.29% to -9.83%, and cross-branch prior integration further improves it to -11.54%. Adding DCNv2-Atten reaches -13.72%. Reducing both latent widths to 160 channels yields $B_0$ at -12.91% while substantially reducing parameter count and decoding time relative to the preceding 128.30 M configuration.

Table III shows that all three insertion locations improve BD-Rate relative to the no-DCNv2 control. The largest individual improvement occurs in the synthesis transform, where the module acts directly on the representations used for reconstruction. The hyperprior and entropy-model insertions provide smaller complementary gains through side-information extraction and conditional probability estimation.

Authorized licensed use limited to: SIMON FRASER UNIVERSITY. Downloaded on November 19,2026 at 23:45:16 UTC from IEEE Xplore. Restrictions apply.
© 2026 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Circuits and Systems for Video Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2026.3742841



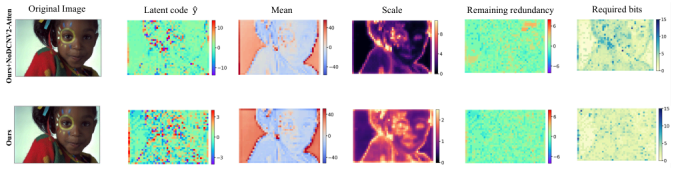



Fig. 9. Representative visualization of entropy-model behavior with and without DCNv2-Atten. The visualization is used as qualitative evidence of altered probability estimation; quantitative coding conclusions are based on the BD-Rate ablations.

TABLE IV
ATTENTION-MODULE COMPARISON ON KODAK FOR THE BASE MODEL $B_0$.

| Attention module | BD-Rate |
|---|---|
| No attention | -10.06% |
| SWAtten [10] | -11.86% |
| DCNv2-Atten | **-12.91%** |

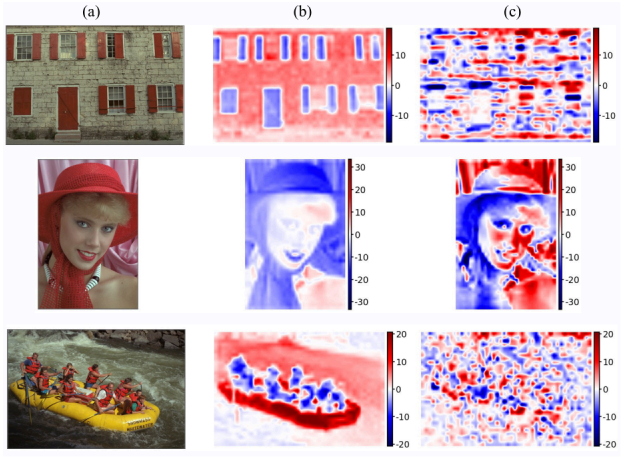



Fig. 10. Intermediate feature maps in the conference $1 \times 1/3 \times 3$ encoder. The visualization illustrates differing response patterns but is not interpreted as an orthogonal frequency decomposition.

### *E. Performance-Oriented Refinement Study*

The following experiments quantify the performance-oriented choices used by the final configuration. They are reported to explain the final R-D improvement and to rule out simple alternative explanations; they are not presented as independent contributions.

The measured sequence improves from $-12.91\%$ for $B_0$ to $-20.23\%$ for the final model $B_4$. Importantly, the transition from $B_0$ to $B_1$ changes both the total number and the spatial placement of TCM blocks; the 1.70-point difference is therefore not attributed solely to stage placement. Table VI separately compares two six-block schedules and isolates the effect of spatial placement within the same total-depth regime. Likewise, the cumulative $B_2 \rightarrow B_3$ change is reported as a practical Stage-II training refinement rather than as an independent methodological contribution.

The nonuniform schedule is 0.46 percentage points better than the equal-depth control. We refer to this as an equal-depth, rather than strictly capacity-matched, comparison because relocating blocks across resolutions can change activation memory and computational cost even when the number of blocks is unchanged.

At fixed latent width, eight groups outperform both five and ten groups. The result shows that the gain is not monotonic in the number of serial channel groups; the selected granularity provides the best measured R-D result among the tested settings. This experiment does not imply equal parameter count across all grouping choices unless the group-specific networks are also parameter-matched.

Under the same Stage-II setting, PU improves Kodak BD-Rate from $-18.33\%$ to $-20.23\%$. At the representative operating point, $R_{FS}$ decreases from $0.188$ to $0.171$ bpp (about $9.0\%$), while the total rate decreases from $0.426$ to $0.408$ bpp (about $4.2\%$). These measurements are consistent with the intended role of Eq. (10): increasing the usefulness of the correctly matched structural condition for detail-stream probability estimation. The BD-Rate result, rather than the single operating point, remains the primary rate–distortion evidence.

### *F. Structural Analysis of the Dual-Branch Representation*

The single-branch controls in Table IX have parameter counts close to the base dual-branch model $B_0$ but yield weaker BD-Rate. This result supports the usefulness of the tested dual-branch factorization within this controlled model family; it does not establish that every possible single-branch architecture is inferior. The mixed-kernel single-stream control is particularly informative because it contains both $3 \times 3$ and $5 \times 5$ processing but does not maintain two separately coded representations.

The three-branch variant also performs worse than the two-branch reference. A pointwise $1 \times 1$ convolution does not aggregate neighboring samples within that layer, although preceding layers can still provide spatial context. The measured three-branch result is therefore interpreted as evidence that adding another branch and a longer conditional chain is not automatically beneficial, rather than as proof that pointwise features are intrinsically uninformative.

Table XI shows that the base model $B_0$ assigns a larger fraction of the coded latent rate to $y_{CS}$ at low rate, while the share of $y_{FS}$ grows as the total rate increases. This trend is consistent with the intended coarse-to-fine coding role, but the bit shares alone do not prove semantic disentanglement. We therefore interpret them together with the controlled architecture comparisons and feature visualizations.

The feature maps in Figs. 10 and 11 provide qualitative support for different branch response patterns. Because kernel size does not enforce exclusive semantic roles, we avoid interpreting these visualizations as proof of strict feature disentanglement. Their value is instead complementary to the quantitative architecture controls and bitrate-allocation measurements.

### *G. Limitations and Future Extensions*

The proposed conditional structure has an explicit decoding dependency. After the hyperlatent, five structural slices must be decoded before the eight detail groups can be decoded conditionally. The finer detail grouping therefore introduces

Authorized licensed use limited to: SIMON FRASER UNIVERSITY. Downloaded on November 19,2026 at 23:45:16 UTC from IEEE Xplore. Restrictions apply.
© 2026 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Circuits and Systems for Video Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2026.3742841



TABLE V
MEASURED REFINEMENT SEQUENCE ON KODAK. $B_0$ IS THE BASE DUAL-BRANCH MODEL USED IN SEC. IV-D. THE SEQUENCE RECORDS THE PRACTICAL MODEL EVOLUTION; IT IS NOT INTERPRETED AS A ONE-VARIABLE CAUSAL DECOMPOSITION. CONTROLLED COMPARISONS BELOW ISOLATE STAGE PLACEMENT, DETAIL-GROUP COUNT, AND PU.

| ID | TCM configuration | $G_{CS}/G_{FS}$ | Stage-II crop | PU | BD-Rate ↓ |
|---|---|---|---|---|---|
| $B_0$ | 3 blocks/path, base placement | 5 / 5 | 384 | No | -12.91% |
| $B_1$ | 6 blocks/path: $\boldsymbol{d}_a = (1,2,3)$, $\boldsymbol{d}_s = (3,2,1)$ | 5 / 5 | 384 | No | -14.61% |
| $B_2$ | same as $B_1$ | 5 / 8 | 384 | No | -16.56% |
| $B_3$ | same as $B_1$ | 5 / 8 | 512 | No | -18.33% |
| $B_4$ final | same as $B_1$ | 5 / 8 | 512 | Yes | **-20.23%** |

TABLE VI
EQUAL-DEPTH CONTROL FOR THE STAGE-WISE TCM SCHEDULE ON KODAK. BOTH SCHEDULES CONTAIN SIX TCM BLOCKS PER PATH; THEIR SPATIAL PLACEMENT DIFFERS.

| Depth vectors $\boldsymbol{d}_a, \boldsymbol{d}_s$ | Total blocks | BD-Rate ↓ |
|---|---|---|
| Uniform: $(2,2,2), (2,2,2)$ | 6 / 6 | -14.15% |
| Nonuniform: $(1,2,3), (3,2,1)$ | 6 / 6 | **-14.61%** |

TABLE VII
DETAIL-LATENT GROUPING CONTROL ON KODAK AT A FIXED 160-CHANNEL LATENT WIDTH.

| Groups | Channels/group | BD-Rate ↓ |
|---|---|---|
| 5 | 32 | -14.61% |
| 8 | 20 | **-16.56%** |
| 10 | 16 | -16.42% |

TABLE VIII
EFFECT OF PU UNDER THE SAME STAGE-II HIGH-RESOLUTION TRAINING SETTING ON KODAK. THE BITRATE COLUMNS CORRESPOND TO THE SAME REPRESENTATIVE OPERATING POINT.

| Stage-II objective | $R_{FS}$ (bpp) | Total (bpp) | BD-Rate ↓ |
|---|---|---|---|
| Without PU | 0.188 | 0.426 | -18.33% |
| With PU | **0.171** | **0.408** | **-20.23%** |

TABLE IX
COMPARISON WITH PARAMETER-COMPARABLE SINGLE-BRANCH BASELINES AND MULTI-BRANCH VARIANTS ON KODAK.

| Architecture | Kernels | Dec. | BD-Rate | #Params |
|---|---|---|---|---|
| Single branch (wider) | $3\times3$ | 0.138 s | -11.65% | 92.10 M |
| Single branch (deeper) | $3\times3$ | 0.155 s | -11.82% | 92.55 M |
| Single branch (mixed) | $3\times3 \parallel 5\times5$ | 0.176 s | -12.15% | 92.35 M |
| 2-branch | $1\times1 \parallel 3\times3$ | **0.128 s** | -10.68% | **80.48 M** |
| 3-branch | $1\times1 \parallel 3\times3 \parallel 5\times5$ | 0.286 s | -11.23% | 120.98 M |
| Base model $B_0$ | $3\times3 \parallel 5\times5$ | 0.134 s | **-12.91%** | 91.86 M |

TABLE X
AVERAGE BITRATE ALLOCATION IN THE THREE-BRANCH VARIANT.

| Total bpp | $5\times5$ branch | $3\times3$ branch | $1\times1$ branch |
|---|---|---|---|
| 0.152 | 28.2% | 44.5% | 27.3% |
| 0.470 | 38.5% | 36.0% | 25.5% |
| 0.896 | 42.4% | 34.2% | 23.4% |

TABLE XI
AVERAGE BIT ALLOCATION BETWEEN THE STRUCTURAL AND DETAIL LATENTS OF THE BASE MODEL $B_0$ ON KODAK.

| Operating region | Total bpp | $y_{CS}$ share | $y_{FS}$ share |
|---|---|---|---|
| Low rate | 0.146 | 86.8% | 13.2% |
| Medium rate | 0.450 | 70.2% | 29.8% |
| High rate | 0.867 | 58.9% | 41.1% |

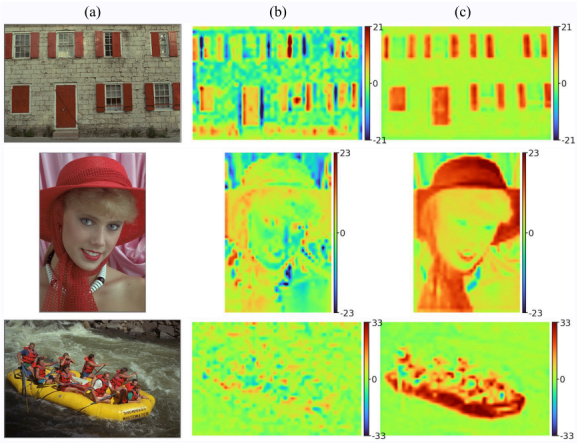



Fig. 11. Intermediate feature maps in the base model $B_0$. The $5\times5$ branch shows stronger responses on broader structures and smooth regions, whereas the $3\times3$ branch responds more strongly to local edges and texture-rich regions in the shown examples.

more serial group-level steps than the five/five base configuration $B_0$. This tradeoff can be unfavorable in applications that prioritize minimum entropy-decoding latency over R-D efficiency.

The structural condition and previously decoded detail contexts must also remain available during detail decoding, which increases feature-memory pressure compared with fully independent streams. Accordingly, runtime and complexity must be evaluated for the final configuration itself rather than transferred from $B_0$, because the final transform depth and entropy-group schedule change the deployed computation.

PU is most useful when the structural latent is predictive of the detail distribution. Highly stochastic local textures can weaken this dependency, and a larger auxiliary weight can alter the optimization tradeoff. PU is therefore used as a moderate auxiliary term rather than a replacement for the likelihood objective. The current codec also trains separate checkpoints

Authorized licensed use limited to: SIMON FRASER UNIVERSITY. Downloaded on November 19,2026 at 23:45:16 UTC from IEEE Xplore. Restrictions apply.
© 2026 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Circuits and Systems for Video Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2026.3742841



for different Lagrange multipliers. A variable-rate extension could condition the transforms and entropy model on a rate-control embedding while preserving the same coarse-to-fine decoding order; this remains future work.

## V. Conclusions

We presented a dual-branch learned image compression framework that couples receptive-field specialization with an explicit coarse-to-fine entropy-coding dependency. The structural latent is decoded first and provides both value and distributional conditions for coding the detail latent. DCNv2-Atten supplies content-adaptive aggregation in the synthesis, hyperprior-analysis, and entropy-parameter paths. The prior-utilization regularizer further encourages the conditional probability model to derive a measurable rate benefit from the correctly matched structural prior, without changing the deployed entropy factorization.

The final model achieves BD-Rate reductions of $-20.23\%$, $-23.58\%$, and $-23.46\%$ on Kodak, Tecnick, and CLIC Professional Validation, respectively, and achieves a lower BD-Rate than the strongest competing entry in Table I on all three datasets. Controlled experiments separately examine stage-wise transform depth, detail-latent grouping, high-resolution fine-tuning, and PU. The first three are treated as performance-oriented implementation refinements rather than independent contributions; the scientific claim remains centered on the coarse-to-fine representation/entropy dependency, adaptive cross-branch aggregation, and explicit prior-utilization objective.

## References


[1] G. K. Wallace, "The JPEG still picture compression standard," *IEEE Transactions on Consumer Electronics*, vol. 38, no. 1, pp. 18–34, 1992.

[2] A. Skodras, C. Christopoulos, and T. Ebrahimi, "The JPEG 2000 still image compression standard," *IEEE Signal Processing Magazine*, vol. 18, no. 5, pp. 36–58, 2001.

[3] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1649–1668, 2012.

[4] B. Bross, Y.-K. Wang, Y. Ye, S. Liu, J. Chen, G. J. Sullivan, and J.-R. Ohm, "Overview of the versatile video coding (VVC) standard and its applications," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 10, pp. 3736–3764, 2021.

[5] J. Ballé, V. Laparra, and E. P. Simoncelli, "End-to-end optimized image compression," in *International Conference on Learning Representations (ICLR)*, 2017.

[6] J. Ballé, D. Minnen, S. Singh, S. J. Hwang, and N. Johnston, "Variational image compression with a scale hyperprior," in *International Conference on Learning Representations (ICLR)*, 2018, pp. 1–23.

[7] D. Minnen, J. Ballé, and G. D. Toderici, "Joint autoregressive and hierarchical priors for learned image compression," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2018, pp. 10 794–10 803.

[8] Z. Cheng, H. Sun, M. Takeuchi, and J. 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 (CVPR)*, 2020, pp. 7939–7948.

[9] H. Fu, F. Liang, J. Liang, B. Li, G. Zhang, and J. Han, "Asymmetric learned image compression with multi-scale residual block, importance scaling, and post-quantization filtering," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 33, no. 8, pp. 4309–4321, 2023.

[10] J. Liu, H. Sun, and J. Katto, "Learned image compression with mixed transformer–CNN architectures," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 14 388–14 397.

[11] D. Minnen and S. Singh, "Channel-wise autoregressive entropy models for learned image compression," in *2020 IEEE International Conference on Image Processing (ICIP)*, 2020, pp. 3339–3343.

[12] 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," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 5718–5727.

[13] D. Li, Y. Bai, K. Wang, J. Jiang, X. Liu, and W. Gao, "Groupedmixer: An entropy model with group-wise token-mixers for learned image compression," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 34, no. 10, pp. 9606–9619, 2024.

[14] D. Feng, Z. Cheng, S. Wang, R. Wu, H. Hu, G. Lu, and L. Song, "Linear attention modeling for learned image compression," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025, pp. 7623–7632.

[15] F. Zeng, H. Tang, Y. Shao, S. Chen, L. Shao, and Y. Wang, "MambaIC: State space models for high-performance learned image compression," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025, pp. 18 041–18 050.

[16] Y. Li, H. Zhang, L. Li, and D. Liu, "Learned image compression with hierarchical progressive context modeling," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025, pp. 18 834–18 843.

[17] J. Wang and Q. Ling, "FDNet: Frequency decomposition network for learned image compression," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 34, no. 11, pp. 11 241–11 255, 2024.

[18] H. Fu, F. Liang, J. Liang, Z. Fang, G. Zhang, and J. Han, "Learned image compression with dual-branch encoder and conditional information coding," in *2024 Data Compression Conference (DCC)*, 2024, pp. 173–182.

[19] J. Ballé, V. Laparra, and E. P. Simoncelli, "End-to-end optimization of nonlinear transform codes for perceptual quality," in *2016 Picture Coding Symposium (PCS)*, 2016, pp. 1–5.

[20] H. Fu, F. Liang, J. Lin, B. Li, M. Akbari, J. Liang, G. Zhang, D. Liu, C. Tu, and J. Han, "Learned image compression with gaussian–laplacian–logistic mixture model and concatenated residual modules," *IEEE Transactions on Image Processing*, vol. 32, pp. 2063–2076, 2023.

[21] Y. Xie, K. L. Cheng, and Q. Chen, "Enhanced invertible encoding for learned image compression," in *Proceedings of the ACM International Conference on Multimedia*, 2021, pp. 162–170.

[22] D. Mishra, S. K. Singh, and R. K. Singh, "Wavelet-based deep auto encoder–decoder (WDAED)-based image compression," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 4, pp. 1452–1462, 2021.

[23] H. Li, S. Li, W. Dai, C. Li, J. Zou, and H. Xiong, "Frequency-aware transformer for learned image compression," in *International Conference on Learning Representations (ICLR)*, 2024. [Online]. Available: https://openreview.net/forum?id=HKGQDDTuvZ

[24] D. He, Y. Zheng, B. Sun, Y. Wang, and H. Qin, "Checkerboard context model for efficient learned image compression," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 14 771–14 780.

[25] W. Jiang and R. Wang, "MLIC++: Linear complexity multi-reference entropy modeling for learned image compression," in *The 40th International Conference on Machine Learning (ICML) Workshop*, 2023.

[26] M. Han, S. Jiang, S. Li, X. Deng, M. Xu, C. Zhu, and S. Gu, "Causal context adjustment loss for learned image compression," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 37, 2024, pp. 133 231–133 253.

[27] T. Chen, H. Liu, Z. Ma, Q. Shen, X. Cao, and Y. Wang, "End-to-end learnt image compression via non-local attention optimization and improved context modeling," *IEEE Transactions on Image Processing*, vol. 30, pp. 3179–3191, 2021.

[28] A. B. Koyuncu, H. Gao, A. Boev, G. Gaikov, E. Alshina, and E. Steinbach, "Contextformer: A transformer with spatio-channel attention for context modeling in learned image compression," in *Computer Vision – ECCV 2022*, 2022, pp. 447–463.

[29] X. Zhu, H. Hu, S. Lin, and J. Dai, "Deformable convnets v2: More deformable, better results," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[30] J. Liu, D. Liu, W. Yang, S. Xia, X. Zhang, and Y. Dai, "A comprehensive benchmark for single image compression artifact reduction," *IEEE Transactions on Image Processing*, vol. 29, pp. 7845–7860, 2020.

[31] "Kodak PhotoCD Dataset," Online. [Online]. Available: http://r0k.us/graphics/kodak/


Authorized licensed use limited to: SIMON FRASER UNIVERSITY. Downloaded on November 19,2026 at 23:45:16 UTC from IEEE Xplore. Restrictions apply.
© 2026 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Circuits and Systems for Video Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2026.3742841



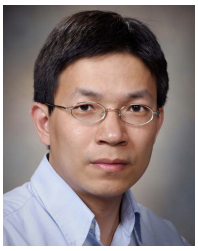
[32] N. Asuni and A. Giachetti, "TESTIMAGES: A large-scale archive for testing visual devices and basic image processing algorithms," in *STAG: Smart Tools and Apps for Graphics*. The Eurographics Association, 2014.
[33] G. Toderici, R. Timofte, J. Ballé, E. Agustsson, N. Johnston, and F. Mentzer, "2021 workshop and challenge on learned image compression (CLIC)," CVPR Workshop and Challenge on Learned Image Compression, 2021. [Online]. Available: http://www.compression.cc
[34] Y. Hu, W. Yang, Z. Ma, and J. Liu, "Learning end-to-end lossy image compression: A benchmark," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2021.
[35] Y. Zhu, Y. Yang, and T. Cohen, "Transformer-based transform coding," in *International Conference on Learning Representations (ICLR)*, 2022.
[36] R. Zou, C. Song, and Z. 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 (CVPR)*, 2022, pp. 17 492–17 501.
[37] Y. Qian, M. Lin, X. Sun, Z. Tan, and R. Jin, "Entroformer: A transformer-based entropy model for learned image compression," in *International Conference on Learning Representations (ICLR)*, 2022.
[38] H. Fu, J. Liang, Z. Fang, J. Han, F. Liang, and G. Zhang, "Weconvene: Learned image compression with wavelet-domain convolution and entropy model," 2024. [Online]. Available: https://arxiv.org/abs/2407.09983
[39] Fraunhofer HHI, "VVC official test model (VTM)," Online, 2019. [Online]. Available: https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware_VTM



**Haisheng Fu** (Student Member, IEEE) received the Ph.D. degree in electronic science and technology from Xi'an Jiaotong University, Xi'an. During his Ph.D. studies, he spent two years as a joint Ph.D. student at Simon Fraser University, Canada. He is currently a postdoctoral fellow at Simon Fraser University. His research interests include Machine Learning, Image and Video Compression, Deep Learning, and VLSI Design. He is also an active reviewer for several prestigious journals and conferences, including IEEE TPAMI, IEEE TIP, IEEE TCSVT, ICASSP, and ICIP.



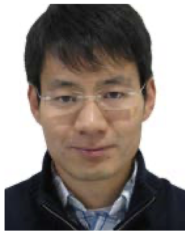
**Feng Ding** received the M.Sc. degree in the School of Information Science and Engineering from Shandong Normal University, Jinan, China, in 2023. He is currently pursuing the Ph.D. degree with the School of Engineering Science, Simon Fraser University Canada. His research interests include Action Recognition and Image Compression.



**Jie Liang** (Senior Member, IEEE) received the B.E. and M.E. degrees from Xi'an Jiaotong University, China, the M.E. degree from National University of Singapore, and the PhD degree from the Johns Hopkins University, USA, in 1992, 1995, 1998, and 2003, respectively. From 2003 to 2004, he worked at the Video Codec Group of Microsoft Digital Media Division. Since May 2004, he has been with the School of Engineering Science, Simon Fraser University, Canada, where he is currently a Professor. Jie Liang's research interests include Image and Video Processing, Computer Vision, and Deep Learning. He had served as an Associate Editor for several journals, including IEEE Transactions on Image Processing, IEEE Transactions on Circuits and Systems for Video Technology (TCSVT), and IEEE Signal Processing Letters. He has also served on three IEEE Technical Committees. He received the 2014 IEEE TCSVT Best Associate Editor Award, 2014 SFU Dean of Graduate Studies Award for Excellence in Leadership, and 2015 Canada NSERC Discovery Accelerator Supplements (DAS) Award.



**Zhenman Fang** (Member, IEEE) received the Ph.D. degree in computer science from Fudan University, Shanghai, China, in 2014. He conducted postdoctoral research at the University of California, Los Angeles (UCLA), Los Angeles, CA, USA, from 2014 to 2017, and worked as a Staff Software Engineer at Xilinx, San Jose, CA, USA, from 2017 to 2019. He is currently an Assistant Professor with the School of Engineering Science, Simon Fraser University, Burnaby, BC, Canada. His recent research focuses on customizable computing with specialized hardware acceleration, including emerging application characterization and acceleration, novel accelerator-rich and near-data computing architecture designs, and corresponding programming, runtime, and tool support.



**Jingning Han** (Senior Member, IEEE) received the B.S. degree in electrical engineering from Tsinghua University, Beijing, China, in 2007, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of California at Santa Barbara, Santa Barbara, CA, USA, in 2008 and 2012, respectively. He joined the WebM Codec Team, Google, Mountain View, CA, USA, in 2012, where he is the Main Architect of the VP9 and AV1 codecs, and leads the Software Video Codec Team. He has published more than 60 research articles. He holds more than 50 U.S. patents in the field of video coding. His research interests include video coding and computer science architecture. Dr. Han received the Dissertation Fellowship from the Department of Elec- trical and Engineering, University of California at Santa Barbara, in 2012. He was a recipient of the Best Student Paper Award at the IEEE International Conference on Multimedia and Expo, in 2012. He also received the IEEE Signal Processing Society Best Young Author Paper Award, in 2015.

Authorized licensed use limited to: SIMON FRASER UNIVERSITY. Downloaded on November 19,2026 at 23:45:16 UTC from IEEE Xplore. Restrictions apply.
© 2026 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.