PO-RTIntra: Perception-Weighted Rate Allocation and ROI Latent Optimization on DCVC-RT Intra

Wenzhuo Ma, Junxi Zhang, Nianxiang Fu, Zhenzhong Chen*

School of Remote Sensing and Information Engineering, Wuhan University, Wuhan, China

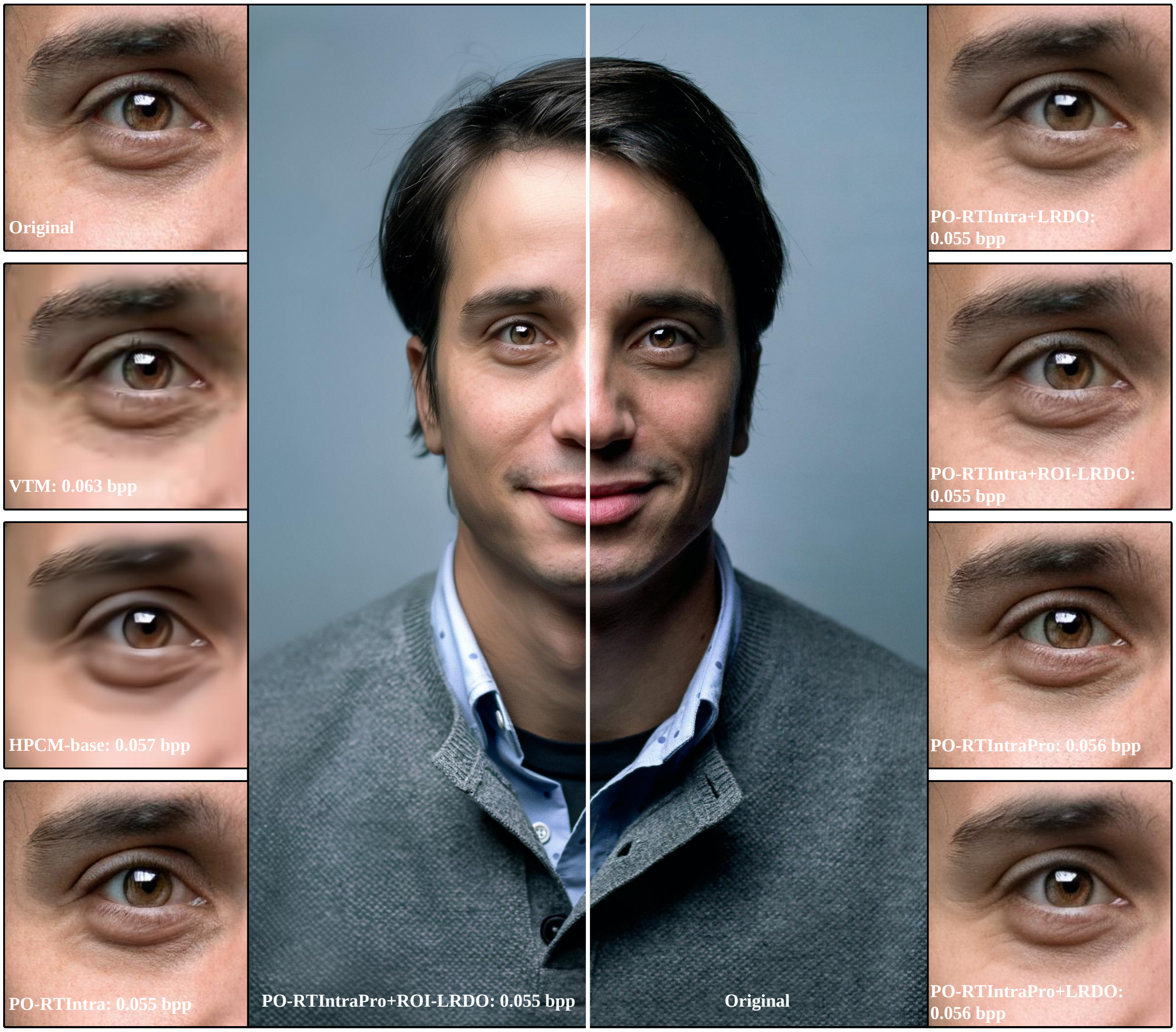
Original
VTM: 0.063 bpp
HPCM-base: 0.057 bpp
PO-RTIntra: 0.055 bpp
PO-RTIntraPro+ROI-LRDO: 0.055 bpp
Original
PO-RTIntra+LRDO: 0.055 bpp
PO-RTIntra+ROI-LRDO: 0.055 bpp
PO-RTIntraPro: 0.056 bpp
PO-RTIntraPro+LRDO: 0.056 bpp

Fig. 1: Visual comparison between the proposed method and baseline methods.

***Abstract*—In this paper, we propose PO-RTIntra, a perception-oriented learned image compression framework built upon the DCVC-RT intra model, together with a higher-capacity variant, PO-RTIntraPro. The strong compression performance and efficiency of the DCVC-RT intra model provide a solid backbone for PO-RTIntra, while PO-RTIntraPro increases the capacity of key modules to further enhance modeling capacity. We adopt a multi-stage progressive training schedule and incorporate a composite perceptual loss together with a Relativistic PatchGAN discriminator to improve perceptual fidelity. In addition, we introduce a Human-Perception-weighted Integer Linear Programming (HP-ILP) formulation for bitrate allocation, and an ROI-based Latent Rate–Distortion-Optimized (ROI-LRDO) inference strategy to further improve reconstruction quality. Experiments demonstrate that, compared with state-of-the-art image compression methods, our approach produces more realistic, detail-rich reconstructions.**

***Index Terms*—Learned image compression, perceptual quality**

* Corresponding author: Zhenzhong Chen (Email: zzchen@whu.edu.cn).

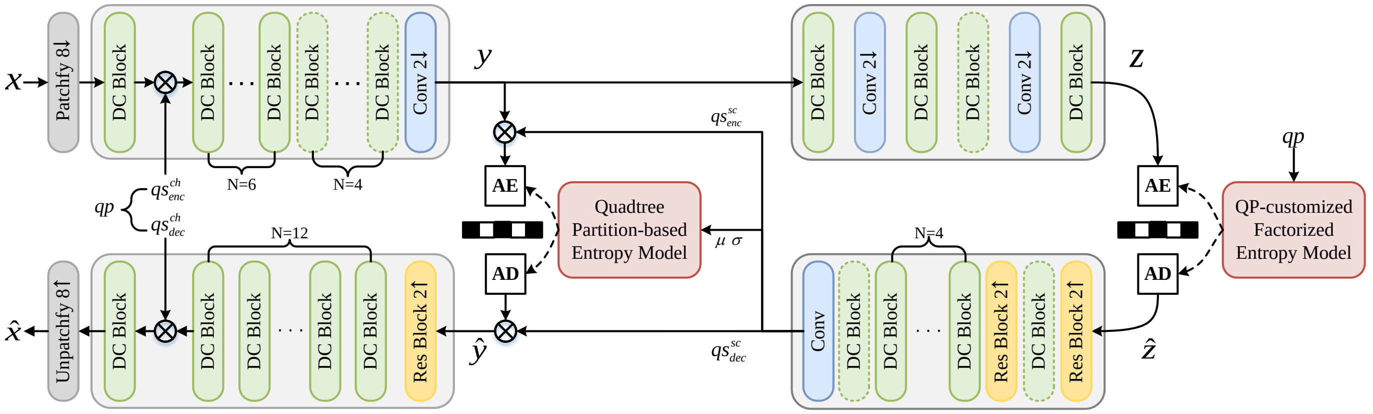

Fig. 2: Network architectures of the proposed PO-RTIntra and PO-RTIntraPro. Dashed-line blocks indicate components exclusive to PO-RTIntraPro. DC Block indicates the depth-wise convolution block from DCVC-RT [3].

I. Introduction

Image compression is fundamental to the efficient storage and transmission of visual data. Traditional codecs (e.g., HEVC [1], VVC [2]) deliver strong rate–distortion performance, yet their modular, hand-engineered pipelines limit joint optimization across components and adapt poorly to diverse image content. Moreover, these systems are typically tuned for objective fidelity measures such as PSNR, which are often misaligned with human visual perception. In contrast, learning-based image compression enables end-to-end optimization, flexible content adaptivity, and the direct incorporation of perceptual objectives. Building on the intra-frame model from recent work [3], we propose a perception-oriented framework, PO-RTIntra, along with its high-capacity variant, PO-RTIntraPro. To enhance perceptual fidelity, we introduce a multi-stage progressive training schedule that combines a composite perceptual loss with a Relativistic Patch-GAN discriminator. Beyond training, we design a Human-Perception-weighted Integer Linear Programming (HP-ILP) strategy for content-aware bitrate allocation and a Region-of-Interest Latent Rate–Distortion–Optimized (ROI-LRDO) inference procedure that emphasizes visually critical regions without sacrificing global coherence. Extensive experiments show that our approach produces reconstructions that are more visually realistic and richer in detail than state-of-the-art learned and conventional codecs at comparable bitrates.

II. Methodology

In this section, we first present the architectures of the proposed PO-RTIntra and PO-RTIntraPro models and clarify their differences. We then describe the multi-stage progressive training schedule and our carefully designed loss functions. Finally, we introduce the human-perception-weighted integer linear programming algorithm for rate allocation and the ROI-based latent rate–distortion–optimized inference strategy.

A. Network Architecture

Our approach is built upon the DCVC-RT intra model [3], chosen for its strong compression performance and efficiency, and its network architecture is shown in Fig. 2. Specifically, on the encoder side, the input image x is first downsampled by a factor of 8 via a Patchfy operation, and then transformed by the main encoder g_a into a compact latent representation y. Quantization and entropy coding then produce the bitstream. On the decoder side, the reconstructed latent $\hat{y}$ is obtained by entropy-decoding the bitstream and then passed through the main decoder g_s followed by an Unpatchfy operation to produce the reconstructed image $\hat{x}$. The latent y and the hyperprior z are modeled using a Quadtree Partition–based Entropy Model and a QP-customized Factorized Entropy Model, respectively. In addition, the proposed models support variable-rate capability, adopting a variable-rate scheme similar to DCVC-FM [4].

We introduce two architectures with similar designs but different complexities to participate in the two competition tracks: PO-RTIntra (low complexity, for the CPU track) and PO-RTIntraPro (high complexity, for the GPU track). Specifically, PO-RTIntra is nearly identical to the DCVC-RT intra backbone, except that the number of supported quantization parameter (QP) points per model is reduced from 64 to 24. PO-RTIntraPro builds on PO-RTIntra by increasing the number of DC blocks and the widths of intermediate channels in both the main transform and the hyperprior transform networks, thereby increasing model capacity. The overall architectural differences, computational complexity, and decoding time for PO-RTIntra and PO-RTIntraPro are summarized in Table I.

B. Training Schedule

To fully train the model and obtain high-perceptual-quality reconstructions, we adopt a multi-stage progressive training schedule. An overall schematic is shown in Fig. 3, and per-stage details are summarized in Table II.

Stage 1 (basic capability). We endow the model with fundamental compression–reconstruction capability and variable-rate control. Training is first conducted at the highest bitrate point and then continued across all bitrate points. The objective is the standard rate–distortion loss in Eq. 1, where R denotes the rate term and λ balances rate and distortion:

$$L_1 = R + \lambda MSE(x, \hat{x}) \tag{1}$$

TABLE I: Comparison of Architecture, Complexity, and Decoding Time between PO-RTIntra and PO-RTIntraPro.

Model	Track	Number of DC Blocks				Number of Channels						Complexity		Dec Time (s)	
		g_a	g_s	h_a	h_s	g_a	g_s	h_a	h_s	y	z	MACs (K)	Params (M)	g_s (fp32)	g_s (fp16)
PO-RTIntra	CPU	7	13	1	4	368	368	128	128	256	128	486.4	45.5	25	19
PO-RTIntraPro	GPU	11	13	2	6	512	512	192	192	256	192	948.4	78.7	32	22

[*] Dec Time (s) denotes the total time, in seconds, to decode the CLIC 2025 Image Validation dataset (32 2K images) on a single NVIDIA L4 GPU.

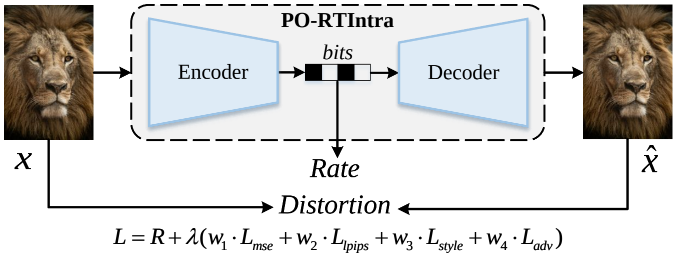
PO-RTIntra
Encoder
bits
Decoder
x
$\hat{x}$
Rate
Distortion
$L = R + \lambda(w_1 \cdot L_{mse} + w_2 \cdot L_{lpips} + w_3 \cdot L_{style} + w_4 \cdot L_{adv})$

Fig. 3: The overall schematic of the multi-stage progressive training schedule.

Stage 2 (compression performance). To ensure strong compression performance, we continue training across all bitrate points with the same objective in Eq. 1, while increasing the patch size and applying a gradually decaying learning rate.

Stage 3 (perceptual fidelity). To improve perceptual quality, we train over all bitrate points using a composite perceptual loss that combines MSE and LPIPS [5], as in Eq. 2. The weight w_2 is normalized to match the scale of the MSE term:

$$L_2 = R + \lambda(w_1 MSE(x, \hat{x}) + w_2 VGG(x, \hat{x})) \qquad (2)$$

Stage 4 (sharpness and detail). To obtain sharp, detail-rich reconstructions, we continue training across all bitrate points and augment the objective with the Style loss [6] and a relativistic PatchGAN term [7], as specified in Eq. 3. Here, x denotes the real image, $\hat{x}$ the reconstructed image, $B(\cdot)$ the *BCEWithLogitsLoss*, $O_p(\cdot)$ the discriminator's patch-wise logits output, and $\mathbb{E}[\cdot]$ the mean over all patches.

$$\begin{cases} L_D = \frac{1}{2}\,\mathbb{E}[\mathcal{B}\big(O_p(x) - O_p(\hat{x}),\, 1\big) \\ \quad + \mathcal{B}\big(O_p(\hat{x}) - O_p(x),\, 0\big)] \\ L_G = \mathbb{E}[\mathcal{B}\big(O_p(\hat{x}) - O_p(x),\, 1\big)] \\ L_3 = R + \lambda(w_1 MSE(x, \hat{x}) + w_2 VGG(x, \hat{x}) \\ \quad + w_3 Style(x, \hat{x}) + w_4 L_G) \end{cases} \qquad (3)$$

TABLE II: Detailed settings for the multi-stage progressive training schedule.

Stage	Epoch	PS	QP	Loss	w_1	w_2	w_3	w_4
1	60	256	23	L_1	1.0	0.0	0.0	0.0
	40	256	0-23	L_1	1.0	0.0	0.0	0.0
2	80	512	0-23	L_1	1.0	0.0	0.0	0.0
3	60	512	0-23	L_2	0.5	0.5	0.0	0.0
4	60	512	0-23	L_3	0.3	225	1000	300

[*] PS denotes the patch size; QP denotes the set of trainable bitrate points.

C. Inference Strategy

To achieve the best overall perceptual quality under a target-bitrate constraint, we design an inference pipeline (Fig. 4) comprising a human-perception-weighted integer linear programming (HP-ILP) rate-allocation algorithm, an ROI-based latent rate–distortion–optimized (ROI-LRDO) online-inference algorithm, and several engineering refinements (e.g., half-precision decoding and pre-/post-resampling). The details are as follows:

HP-ILP. The task is to determine a per-image bitrate allocation that maximizes overall perceptual quality under a prescribed bitrate budget. Enforcing such a constraint with a model trained with a single λ is often difficult on content-diverse datasets. Fortunately, our models are variable-rate, allowing us to cast the problem as a constrained optimization. A key challenge, however, is choosing a metric that faithfully reflects human visual quality. Prior work [8] uses LPIPS as a perceptual proxy in a linear program, but LPIPS does not perfectly align with human perception—and, to date, no single metric fully captures subjective quality. To bridge this gap, we propose a human-perception-weighted integer linear program (HP-ILP) that augments the LPIPS-based objective with human-perception priors, thereby improving overall subjective quality at the target bitrate. Let $\pi_i^{qp} := \mathrm{LPIPS}(M(x_i, qp), x_i)$ and $r_i^{qp} := R(M(x_i, qp), x_i)$ denote the precomputed LPIPS and bitrate (e.g., in bpp) for image i at bitrate point qp. The HP-ILP is

$$\begin{aligned} \min_{\{f_i^{qp}\}} \quad & \sum_{i=1}^{N} \sum_{qp=0}^{QP-1} w_i\, \pi_i^{qp}\, f_i^{qp} \\ \text{s.t.} \quad & \frac{1}{N} \sum_{i=1}^{N} \sum_{qp=0}^{QP-1} r_i^{qp}\, f_i^{qp} \;\le\; T, \\ & \sum_{qp=0}^{QP-1} f_i^{qp} \;=\; 1, \quad \forall i \in \{1, \dots, N\}, \\ & f_i^{qp} \in \{0, 1\}, \quad \forall i, qp. \end{aligned} \qquad (4)$$

Here, N and QP are the numbers of images and supported bitrate points, respectively; x_i is the i-th image; $M(\cdot)$ is the proposed model; f_i^{qp} indicates whether the qp-th bitrate is selected for image i; and T is the target bitrate (in this challenge, one of $\{0.075, 0.15, 0.3\}$). Finally, $w_i > 0$ is the human-perception weight for image i obtained from subjective assessment. For example, if increasing the bitrate for image i yields a substantial improvement in perceived quality while the LPIPS reduction is small, then $w_i > 1$; conversely, $w_i < 1$ when LPIPS overstates the perceptual improvement.

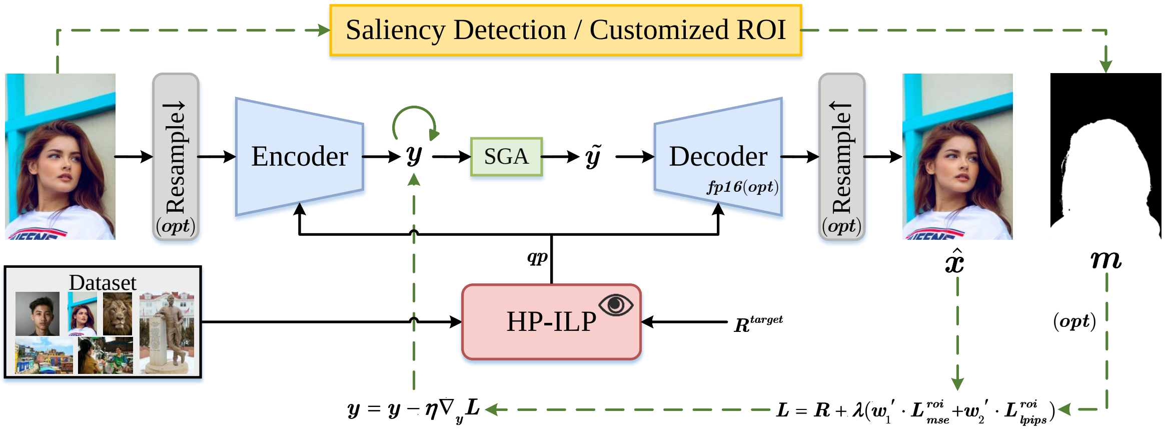

Fig. 4: Inference strategy of the proposed method. (opt) denotes the optional operation.

Algorithm 1: ROI-based latent rate-distortion-optimized inference strategy.

```
Input  : Input image $x$; encoder $\mathcal{E}$; decoder $\mathcal{D}$;
         saliency detector $\mathcal{S}$; arithmetic encoder $\mathcal{AE}$
Output: Reconstructed image $\hat{x}$; bitstream $bits$
$\alpha_r,\ \alpha_n\ (\alpha_r \gg \alpha_n)$: Coefficients for ROI/non-ROI;
$N$: RDO iterations;
$m, y_0 \leftarrow \mathcal{S}(x), \mathcal{E}(x)$;
for $i \leftarrow 0$ to $N-1$ do
    $\tilde{y}_i \leftarrow \mathrm{SGA}(y_i)$;
    $\hat{x}_i \leftarrow \mathcal{D}(\tilde{y}_i)$;
    $R_i \leftarrow \mathcal{R}(\tilde{y}_i)$;
    $mse_i \leftarrow MSE(\hat{x}_i, x)$;
    $lpips_i \leftarrow VGG(\hat{x}_i, x)$;
    $mse_i^{roi} \leftarrow \alpha_r \cdot m \cdot mse_i + \alpha_n \cdot (1-m) \cdot mse_i$;
    $lpips_i^{roi} \leftarrow \alpha_r \cdot m \cdot lpips_i + \alpha_n \cdot (1-m) \cdot lpips_i$;
    $L_i \leftarrow R_i + \lambda(w_1' \cdot mse_i^{roi} + w_2' \cdot lpips_i^{roi})$;
    $y_{i+1} \leftarrow y_i - \eta\, \nabla_{y_i} L_i$;
end
$\hat{x},\ bits \leftarrow \mathcal{D}(round(y_N)),\ \mathcal{AE}(round(y_N))$;
```

ROI-LRDO. Due to the amortization gap, a model trained on large-scale data does not necessarily yield an optimal solution for any single image, particularly on content-diverse test sets [9]. Moreover, human visual attention varies across regions within an image (e.g., in Fig. 4, faces tend to attract more attention than the background). To compensate for this amortization gap and improve intra-image bitrate allocation, we propose an ROI-based latent rate-distortion-optimized online inference algorithm, as shown in Algorithm 1. Specifically, we first obtain an ROI mask m for the input image x using a saliency detection method (e.g., RMFormer [10]) or manual annotation. We then optimize the latent representation via Stochastic Gumbel Annealing (SGA) [9] under the objective in Eq. 5, where w_1' and w_2' are the weights of the distortion terms, and L_{mse}^{roi} and L_{lpips}^{roi} denote the MSE and LPIPS losses computed with pixel-wise weighting by the ROI mask:

$$L = R + \lambda(w_1' \cdot L_{mse}^{roi} + w_2' \cdot L_{lpips}^{roi}) \tag{5}$$

Engineering Refinements. Beyond the HP-ILP and ROI-LRDO algorithms described above, we introduce several engineering refinements to improve overall perceptual quality and accelerate decoding. These include *half-precision decoding* and *pre-/post-resampling*. For half-precision decoding, we cast computations in the decoder g_s from float32 to float16, yielding substantial speedups with negligible impact on both objective metrics and perceived quality. As shown in Table I, half-precision decoding reduces the time to decode the CLIC 2025 image validation set (32 2K resolution images) by 6 s for PO-RTIntra and by 10 s for PO-RTIntraPro. For pre-/post-resampling, given an input image, we first scale it by a factor β in height and width (e.g., $\beta = 0.9$), run the inference pipeline to obtain the selected bitrate and reconstruction, and finally upsample the result back to the original resolution. This technique is particularly effective for images with cluttered fine textures—which often consume many bits—since mild resampling can save bits for other images without perceptual degradation.

III. Experiments

A. Experimental Setting

Training. We construct the training set with a total of 105,899 images drawn from the ImageNet validation set [11], the CLIC 2020 training set [12], DIV2K [13], Flickr2K [14], Flicker2W [15], and the first six shards of LSDIR [16]. We employ the Adam optimizer [17] to minimize the rate-distortion loss. For the learning rate, we keep 1×10^{-4} fixed in Stage 1; in the remaining stages it is initialized at 1×10^{-4} and decayed to 1×10^{-6}. Stage-wise loss weights and other training hyperparameters are summarized in Table II. For PO-RTIntra and PO-RTIntraPro, we train three models each to cover the three target bitrate points; the corresponding λ ranges are $[0.0003, 0.0022]$, $[0.0015, 0.0100]$, and $[0.0080, 0.0275]$, respectively. All experiments are conducted on NVIDIA GeForce RTX 3090 GPUs.

Evaluation. We evaluate the proposed method on the CLIC 2025 Image Test dataset, which contains 30 diverse 2K images. We assess performance from two perspectives: perceptual metrics and visual quality. For perceptual metrics we report LPIPS [5] and DISTS [18]; as baselines we include the state-

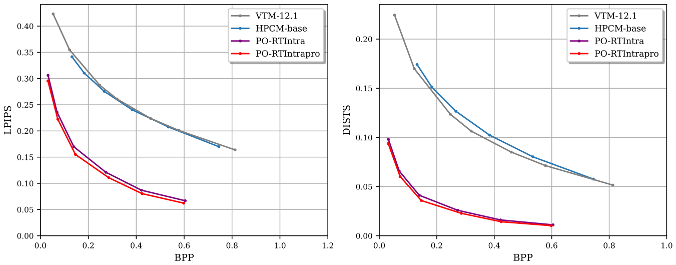

Fig. 5: Rate-distortion curves of the proposed method and baseline methods.

of-the-art conventional codec VTM [19] and the state-of-the-art learned image compression method HPCM [20].

Inference details. In the ROI-LRDO objective, the λ value is aligned with the training-phase λ for the corresponding bitrate (with optional per-image fine-tuning). We set $w_1' = w_2' = 0.5$ and normalize L_{lpips}^{roi} to match the magnitude of L_{mse}^{roi}.For pre-/post-resampling, we use simple bicubic down-/up-sampling to avoid additional decoding time; this technique is applied only at the lowest bitrate for highly textured images (e.g., *9374...3286.png* in the CLIC 2025 Image Test set).

B. Quantitative Results

Fig. 5 shows the perceptual compression performance of our method against baseline methods. Our approach substantially outperforms VTM and HPCM, and PO-RTIntraPro further surpasses PO-RTIntra, owing to its larger model capacity. Notably, the PO-RTIntra and PO-RTIntraPro curves in Fig. 5 were obtained without latent rate-distortion optimization, which would be expected to further improve perceptual metrics.

C. Qualitative Results

Fig. 1 shows the qualitative results of our method against baseline methods. From fine-grained comparisons (e.g., the textures around the eye corners), it is evident that, compared with VTM and HPCM, the proposed method produces sharper, more detail-rich reconstructions. Considering variants of our method (without LRDO, with LRDO, and with ROI-LRDO), LRDO further increases texture detail via online latent-space fine-tuning, whereas ROI-LRDO allocates more bits to the ROI and yields better perceptual quality.

IV. Conclusions

In this paper, we presented PO-RTIntra, a perception-oriented learned image compression framework built upon the DCVC-RT intra model, and its higher-capacity variant PO-RTIntraPro. A multi-stage progressive training schedule, coupled with a composite perceptual loss and a Relativistic PatchGAN discriminator, consistently improves perceptual fidelity. Beyond training, a Human-Perception–weighted Integer Linear Programming (HP-ILP) formulation enables content-aware bitrate allocation, while an ROI-based Latent Rate–Distortion–Optimized (ROI-LRDO) inference strategy further refines visually critical regions. Together with lightweight engineering refinements (e.g., half-precision decoding and pre-/post-resampling), the proposed system achieves strong perceptual quality at comparable bitrates and practical decoding efficiency. Extensive experiments indicate that our approach produces reconstructions that are more realistic and richer in detail than state-of-the-art learned and conventional codecs under the same bitrate constraints.

References

[1] G. J. Sullivan, J. Ohm, W. Han, and T. Wiegand, “Overview of the high efficiency video coding (HEVC) standard,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1649–1668, 2012.

[2] B. Bross, Y. Wang, Y. Ye, S. Liu, J. Chen, G. J. Sullivan, and J. Ohm, “Overview of the versatile video coding (VVC) standard and its applications,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 10, pp. 3736–3764, 2021.

[3] Z. Jia, B. Li, J. Li, W. Xie, L. Qi, H. Li, and Y. Lu, “Towards practical real-time neural video compression,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025, pp. 12 543–12 552.

[4] J. Li, B. Li, and Y. Lu, “Neural video compression with feature modulation,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 26 099–26 108.

[5] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, “The unreasonable effectiveness of deep features as a perceptual metric,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 586–595.

[6] D. He, Z. Yang, H. Yu, T. Xu, J. Luo, Y. Chen, C. Gao, X. Shi, H. Qin, and Y. Wang, “PO-ELIC: perception-oriented efficient learned image coding,” in *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2022, pp. 1763–1768.

[7] A. Jolicoeur-Martineau, “The relativistic discriminator: a key element missing from standard GAN,” in *International Conference on Learning Representations*, 2019.

[8] N. Fu, J. Zhang, H. Wang, and Z. Chen, “Perceptual-oriented learned image compression with dynamic kernel,” in *Data Compression Conference*, 2024, p. 555.

[9] Y. Yang, R. Bamler, and S. Mandt, “Improving inference for neural image compression,” in *Advances in Neural Information Processing Systems*, 2020.

[10] X. Deng, P. Zhang, W. Liu, and H. Lu, “Recurrent multi-scale transformer for high-resolution salient object detection,” in *Proceedings of the ACM International Conference on Multimedia*, 2023, pp. 7413–7423.

[11] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “Imagenet: A large-scale hierarchical image database,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255.

[12] G. Toderici, W. Shi, R. Timofte, L. Theis, J. Balle, E. Agustsson, N. Johnston, and F. Mentzer, “Workshop and challenge on learned image compression,” 2020. [Online]. Available: https://www.tensorflow.org/datasets/catalog/clic

[13] E. Agustsson and R. Timofte, “Ntire 2017 challenge on single image super-resolution: Dataset and study,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2017, pp. 126–135.

[14] R. Timofte, E. Agustsson, L. Van Gool, M.-H. Yang, and L. Zhang, “Ntire 2017 challenge on single image super-resolution: Methods and results,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2017, pp. 114–125.

[15] J. Liu, G. Lu, Z. Hu, and D. Xu, “A unified end-to-end framework for efficient deep image compression,” *arXiv preprint arXiv:2002.03370*, 2020.

[16] Y. Li, K. Zhang, J. Liang, J. Cao, C. Liu, R. Gong, Y. Zhang, H. Tang, Y. Liu, D. Demandolx *et al.*, “Lsdir: A large scale dataset for image restoration,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 1775–1787.

[17] D. P. Kingma and J. L. Ba, “Adam: A method for stochastic optimization,” in *International Conference on Learning Representations*, 2014.

[18] K. Ding, K. Ma, S. Wang, and E. P. Simoncelli, “Image quality assessment: Unifying structure and texture similarity,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 5, pp. 2567–2581, 2022.

[19] “VTM-12.1,” https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware_VTM/, 2022.

[20] Y. Li, H. Zhang, L. Li, and D. Liu, “Learned image compression with hierarchical progressive context modeling,” *arXiv*, vol. abs/2507.19125, 2025.