WIDER: RATE-ADAPTIVE BALANCING FOR WIDE-RANGE RATE–DISTORTION–PERCEPTION CONTROL IN NEURAL IMAGE COMPRESSION

Anonymous authors
Paper under double-blind review

ABSTRACT

Neural image compression has advanced both in reducing distortion and improving perceptual quality, yet improving one can compromise the other at a fixed bitrate. Distortion–perception (D–P) controllable codecs address this trade-off by adjusting the balance between them within a single codec. Existing methods commonly train the codec for rate–distortion performance and then freeze the encoder while adapting the decoder to other preferences. Such adaptation changes how the encoded information is used, but cannot change what is encoded. Our study reveals that distortion-oriented representations incur a perception penalty that grows toward lower rates, whereas perception-oriented representations incur a distortion penalty that grows toward higher rates, even after decoder adaptation. To alleviate these cross-endpoint penalties, we propose rate-adaptive D–P objective balancing, which optimizes the codec with both reconstruction objectives and progressively shifts their relative weighting from perception to distortion as rate increases. We instantiate our method in WIDER, a rate–distortion–perception controllable codec with a pretrained one-step diffusion prior, variable-rate coding, and a decoder-side D–P control mechanism. Across diverse datasets, WIDER demonstrates broader D–P control than EGIC, the strongest baseline among the compared methods, achieving BD-rate gains of up to -60.0% in FID for perception-oriented reconstruction and -5.6% in PSNR for distortion-oriented reconstruction. It also achieves BD-rate gains of up to -20.3% in rate–PSNR at matched FID and -70.6% in rate–FID at matched PSNR. *Code will be released upon publication.*

1 INTRODUCTION

At a given bitrate, lossy image codecs can target different balances between signal distortion and perception, as formalized by rate–distortion–perception (RDP) theory (Blau & Michaeli, 2019). Since the preferred balance varies across applications and users, a straightforward solution is to train and deploy a separate codec for each preference. This approach, however, increases maintenance costs. Universal RDP representation theory (Zhang et al., 2021) studies whether *one shared representation* can support multiple distortion–perception balances. Recent D–P controllable codecs demonstrate such decoder-side control on natural images (Yan et al., 2022; Agustsson et al., 2023; Ghouse et al., 2023; Körber et al., 2024). RDP-controllable codecs additionally vary rate within the same model, as demonstrated by recent works (Iwai et al., 2024; Zhou et al., 2025; Spadaro et al., 2026).

A common approach first optimizes the codec for rate–distortion performance, then freezes the encoder—thereby fixing the compressed representation—and adapts the decoder to support multiple D–P balances (Yan et al., 2022; Körber et al., 2024; Zhou et al., 2025). Such adaptation changes how the representation is used, but not the information it contains. Our study reveals that training the codec with distortion as the sole reconstruction objective produces a distortion-oriented representation (DoR) that constrains the best attainable perceptual quality even after decoder adaptation. Conversely, a perception-oriented representation (PoR) constrains the best attainable distortion. These *cross-endpoint penalties* are rate-dependent: the perception penalty of DoR grows toward lower rates, whereas the distortion penalty of PoR grows toward higher rates (Figure 3).

To alleviate these cross-endpoint penalties, we propose rate-adaptive D–P objective balancing, which shapes the shared compressed representation during codec training by giving perception greater relative emphasis at low rates and distortion greater relative emphasis at high rates. In controlled

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

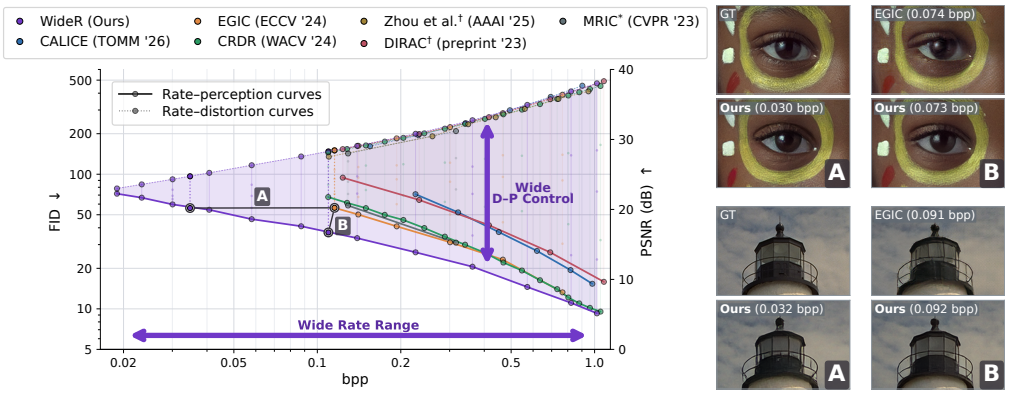


Figure 1: **Comparison of WIDER and existing controllable codecs.** Left: solid and dotted curves show rate–perception and rate–distortion performance, measured by FID (lower is better) and PSNR (higher is better), respectively. The horizontal axis reports bits per pixel (bpp). The proposed model (WIDER) provides the broadest empirical RDP coverage. Right: against the strongest baseline EGIC on kodim15/19, WIDER uses $2.5\times/2.8\times$ fewer bits at matched dataset-level FID (A) and preserves source structure better at matched rate (B). $^{\dagger}/^{*}$ mark digitized/reimplemented results.

experiments using the same architecture, the proposed method reduces the perception penalty of DoR and the distortion penalty of PoR by 68.0% and 90.3%, respectively (Figure 4).

We instantiate this balancing method in WIDER, a wide-range RDP-controllable codec that leverages a pretrained one-step diffusion prior and incorporates a variable-rate latent codec and decoder-side adaptation for D–P control. Evaluations on diverse datasets demonstrate WIDER's advantages over existing controllable codecs in terms of three desirable properties: (i) *wide D–P control*, with endpoint BD-rate gains of up to -60.0% in FID and -5.6% in PSNR; (ii) *better RDP trade-offs*, with BD-rate gains of up to -20.3% in FID-matched rate–PSNR comparisons and -70.6% in PSNR-matched rate–FID comparisons, both evaluated against the strongest baseline, EGIC (Körber et al., 2024); and (iii) *wide rate control*, with D–P control extending into the ultra-low-rate regime.

Our contributions are summarized below.

- We identify cross-endpoint penalties that limit the attainable D–P control range of controllable codecs and propose a rate-adaptive D–P objective balancing method, and we experimentally show that our balancing method substantially alleviates the cross-endpoint penalties.
- We instantiate the proposed balancing method in an RDP-controllable codec, named WIDER, that leverages a one-step diffusion prior. WIDER substantially outperforms existing controllable codecs in D–P control range and in bit savings.
- To the best of our knowledge, WIDER is the *first* RDP-controllable codec that covers, with a single trained model, both the conventional bitrate range (0.1–1.0 bpp) of the controllable codecs and ultra-low-rate range ($\approx$ 0.02 bpp) of the generative realism codecs.

2 Related Work

2.1 Generative Compression

Generative codecs use learned image priors and perceptual objectives to reconstruct plausible detail from compressed signals (Mentzer et al., 2020; Muckley et al., 2023; Careil et al., 2024). Mari & Milani (2024) provide reconstruction flexibility through standard or diffusion decoding of the same compressed latents and through different diffusion sampling procedures, which also change decoding cost. Recent one-step diffusion codecs avoid iterative denoising (Guo et al., 2025; Xue et al., 2025; Zhang et al., 2025; 2026). We also leverage a pretrained one-step diffusion model for high perceptual realism, while incorporating distortion-oriented reconstruction and D–P control within a single codec.

2.2 Controllable Codecs

Rate–distortion–perception (RDP) theory formalizes the effect of a distributional perception constraint on the classical rate–distortion trade-off (Blau & Michaeli, 2019). Subsequent work studies perceptual compression objectives (Yan et al., 2021) and whether one compressed representation can support

multiple distortion–perception balances through adapted decoding (Zhang et al., 2021). Throughout this paper, controllable codecs refers to codecs that support at least D–P control.

D–P controllable codecs. At a fixed rate, these codecs produce multiple distortion–perception balances from a shared bitstream through conditional decoding, endpoint-adapter interpolation, or diffusion-based reconstruction (Yan et al., 2022; Agustsson et al., 2023; Ghouse et al., 2023; Körber et al., 2024). State-of-the-art methods in this group commonly train a separate model for each rate, so their control is over the D–P balances within each rate-specific codec.

RDP-controllable codecs. An RDP-controllable codec additionally controls rate within the same model, requiring one shared variable-rate representation to support both rate and D–P control. CRDR (Iwai et al., 2024) learns a single model for rate, distortion, and realism control. CALICE (Spadaro et al., 2026) uses scalar-conditioned adapters throughout its encoder and decoder to control both rate and D–P balance. Zhou et al. (Zhou et al., 2025) use latent-diffusion decoding to jointly control rate and D–P balance. Across both groups, however, how compressed representation shapes the attainable distortion and perception endpoints remains underexplored.

3 WIDER: WIDE-RANGE RDP-CONTROLLABLE CODEC

3.1 CONTROLLABLE CODEC FORMULATION AND DESIGN GOALS

The rate–distortion–perception (RDP) controllable codec considered in this work comprises an encoder that maps an input image to a compressed representation under a given rate constraint and a decoder that controls the distortion–perception (D–P) balance of the reconstruction from that representation. Let $\mathbf{x}$ be an input image, $q \in [0, 1]$ a normalized rate-control parameter, and $\alpha \in [0, 1]$ a D–P control parameter, which specifies the desired D–P balance. At the representation level, the image compression process of a controllable codec is formulated as

$$(\mathbf{x}, q) \longrightarrow \hat{\mathbf{z}}_q, \qquad (\hat{\mathbf{z}}_q, \alpha) \longrightarrow \hat{\mathbf{x}}_q^{\alpha}, \tag{1}$$

where $\hat{\mathbf{z}}_q$ is the representation compressed under the rate constraint specified by q and is shared across the entire D–P control range. $\hat{\mathbf{x}}_q^{\alpha}$ is the reconstruction from $\hat{\mathbf{z}}_q$ with the D–P balance specified by α. Throughout this paper, q is ordered from low to high rate: $q = 0$ and $q = 1$ denote the lowest and highest supported rate settings, respectively. Likewise, α is oriented from distortion to perception: $\alpha = 0$ and $\alpha = 1$ denote the D and P endpoints, which target the best distortion and perception performance, respectively. Following the convention of the field, D-endpoint performance can be measured by signal-fidelity metrics, whereas P-endpoint performance can be measured by distributional divergences such as FID (Heusel et al., 2017).

The desirable properties of an RDP-controllable codec targeted in this work are: (i) maximizing the attainable D- and P-endpoint performance under a given rate constraint (*wide D–P control*), (ii) maximizing the complementary P performance at a given R–D operating point and D performance at a given R–P operating point (*better RDP trade-off*), and (iii) supporting a broad range of rates (*wide rate control*). Here, wide D–P control refers to achieving both strong distortion and strong perception endpoints from the same compressed representation, rather than merely increasing the difference between endpoint reconstructions. We focus on widening the D–P control range through rate-adaptive D–P objective balancing (Section 3.2). We integrate this method with mechanisms for better RDP trade-offs and wide rate control in a single controllable codec architecture (Section 3.3). We refer to the resulting model as WIDER, a wide-range RDP-controllable codec.

3.2 RATE-ADAPTIVE D–P OBJECTIVE BALANCING

Motivation. We experimentally observe that the prevailing training scheme (Section 1) of controllable codecs— training the codec for rate–distortion performance and then adapting the decoder to other D–P preferences—limits the realism attainable at the P endpoint. Under the same architecture, we train a codec with distortion as the sole reconstruction objective, then adapt the decoder for perception while keeping the resulting distortion-oriented representation (DoR) fixed. Even with the same pretrained one-step diffusion prior, the resulting P endpoint remains substantially behind that of a codec trained with perception as the reconstruction objective, and the shortfall grows toward lower rates (Figure 3). The converse also holds: after decoder adaptation for distortion on a fixed perception-oriented representation (PoR), its distortion performance falls short of that attained from DoR, and this shortfall grows toward higher rates. We refer to this pair of effects as *cross-endpoint penalties*: each objective-specific representation limits the opposite endpoint, and the two limitations dominate at opposite ends of the rate range. These results indicate that the attainable endpoint performance of a

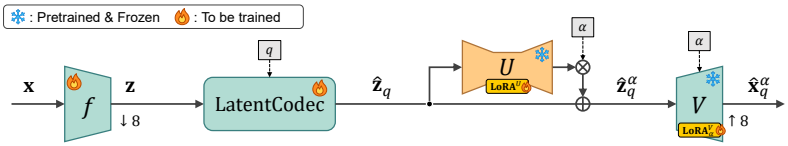


Figure 2: **WIDER architecture.** At rate-control parameter q, a quality-conditioned latent codec produces the shared representation $\hat{\mathbf{z}}_q$. The D–P control parameter α gates the one-step diffusion prior U and mixes endpoint-specific adaptations in the shared VAE decoder V.

controllable codec is constrained by the representation itself. Detailed experimental conditions and additional experiments are described in Section 4.

Methods. Based on these observations, we propose a training objective that adjusts the relative weighting of distortion and perception with rate to alleviate the cross-endpoint penalties:

$$\mathcal{L}(\mathbf{x}, q, \alpha) = \mathcal{L}_R(q) + (1-\alpha)\lambda_D(q)\mathcal{L}_D(\hat{\mathbf{x}}_q^\alpha, \mathbf{x}) + \alpha\lambda_P(q)\mathcal{L}_P(\hat{\mathbf{x}}_q^\alpha, \mathbf{x}), \tag{2}$$

where $\mathcal{L}_R(q)$ is the estimated rate, and $\mathcal{L}_D$ and $\mathcal{L}_P$ are the distortion and perception losses for reconstruction $\hat{\mathbf{x}}_q^\alpha$. We use MSE for $\mathcal{L}_D$, while $\mathcal{L}_P$ combines a lightly weighted MSE term with adversarial and perceptual losses; further details are given in Appendix A.2. At a given rate, α specifies the desired D–P balance, whereas $\lambda_D(q)$ and $\lambda_P(q)$ determine the relative influence of the two reconstruction objectives on the shared representation.

Because the observed cross-endpoint penalties vary in direction and magnitude with rate, we give the perception objective relatively greater influence at low rates and the distortion objective relatively greater influence at high rates by imposing

$$\lambda_D(q_1)/\lambda_P(q_1) > \lambda_D(q_2)/\lambda_P(q_2), \qquad q_1 > q_2. \tag{3}$$

Thus, as rate increases, the relative training pressure on the shared representation shifts progressively from perception toward distortion. To instantiate this design principle, we parameterize the two weighting functions by geometric interpolation:

$$\lambda_D(q) = \left(\lambda_D^{\mathrm{lo}}\right)^{1-q}\left(\lambda_D^{\mathrm{hi}}\right)^{q}, \qquad \lambda_P(q) = \left(\lambda_P^{\mathrm{lo}}\right)^{1-q}\left(\lambda_P^{\mathrm{hi}}\right)^{q}, \tag{4}$$

where the superscripts lo and hi denote the weights at $q=0$ and $q=1$, respectively. This smoothly connects the exponentially scaled low- and high-rate weights through linear interpolation in log space. In practice, we set the boundary weights to $(\lambda_D^{\mathrm{lo}}, \lambda_D^{\mathrm{hi}}, \lambda_P^{\mathrm{lo}}, \lambda_P^{\mathrm{hi}}) = (2^{-9}, 2^{11}, 2^{-4}, 2^{2})$ to satisfy Equation 3 and calibrate the scale of the reconstruction losses relative to the rate term. Consequently, $\lambda_D(q)/\lambda_P(q)$ increases from 2^{-5} at $q=0$ to 2^{9} at $q=1$. These values are selected empirically; their limitations and opportunities for future optimization are discussed in Section 6.

3.3 Architecture and Coding Process

WIDER leverages the generative prior of a pretrained one-step diffusion model (Sauer et al., 2024). A variable-rate mechanism is incorporated for rate control. Its high-level architecture is illustrated in Figure 2 and comprises a latent-space projection f, a variable-rate latent codec, a one-step diffusion U-Net, and a VAE decoder. Given an input image $\mathbf{x} \in \mathbb{R}^{H\times W\times 3}$ and a rate-control parameter q, the compressed representation is obtained as

$$\mathbf{z} = f(\mathbf{x}), \quad \hat{\mathbf{z}}_q = \mathrm{LatentCodec}(\mathbf{z}, q), \tag{5}$$

where f maps the input image to a latent feature map $\mathbf{z} \in \mathbb{R}^{H/8\times W/8\times 256}$. The latent codec comprises analysis and synthesis transforms and an entropy model; it determines the compression level from q and outputs the compressed representation $\hat{\mathbf{z}}_q$ with the same dimensions as $\mathbf{z}$. For the 64 supported rate points, let $i_q \in \{0, \ldots, 63\}$ denote the discrete quality index corresponding to $q = i_q/63$. For rate control, channel-wise scaling by the quality-indexed vectors $M_E[i_q]$ and $M_D[i_q]$ is applied before the analysis transform and after the synthesis transform, respectively (Li et al., 2024). To balance performance and complexity, we use partial-convolution (PConv)-based neural blocks (Chen et al., 2023) in f and the latent-codec transforms, and adopt a quadtree-partition entropy model (Li et al., 2023). Detailed block configurations are provided in Appendix F.1.

The decoder uses a pruned U-Net initialized from SD-Turbo (Sauer et al., 2024) and a half-VAE decoder, with both reduced components following AdcSR (Chen et al., 2025), as in the well-established

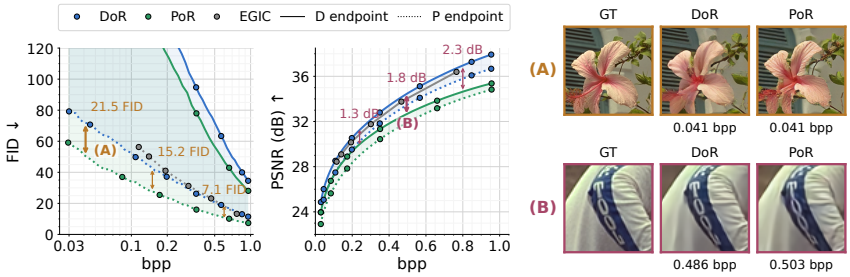


Figure 3: **Cross-endpoint penalties of objective-specific representations.** At matched bpp, the P endpoint of the DoR variant has higher FID than that of the PoR variant, with a larger gap at lower rates (left), while the D endpoint of the PoR variant has lower PSNR than that of the DoR variant, with a larger gap at higher rates (center). The examples in (A) and (B) visually reflect these penalties.

generative codec AEIC-ME (Zhang et al., 2026). Given a compressed representation $\hat{\mathbf{z}}_q$ and a D–P control parameter α, the reconstruction is obtained as

$$\hat{\mathbf{z}}_q^{\alpha} = \hat{\mathbf{z}}_q + \alpha U(\hat{\mathbf{z}}_q), \quad \hat{\mathbf{x}}_q^{\alpha} = V_{\alpha}(\hat{\mathbf{z}}_q^{\alpha}), \tag{6}$$

where U predicts a latent-space residual that introduces the generative prior. Scaling this residual by α continuously steers the reconstruction from the D endpoint toward the P endpoint. V_{α} reconstructs the image from the preference-adjusted latent and internally controls the fine-grained D–P balance according to α. All pretrained parameters of the U-Net and VAE decoder are frozen. Only the U-Net input/output convolutions and rank-32 LoRA adapters (Hu et al., 2022), and the VAE decoder input/output convolutions and rank-16 LoRA adapters, are additionally trained.

We instantiate two endpoint-specific LoRA adapters inside the VAE decoder and mix their outputs according to α. This enables fine-grained D–P control rather than pixel-space mixing. At each layer ℓ,

$$\mathrm{LoRA}_{\alpha,\ell}^{V}(\mathbf{h}) = \left[(1-\alpha)\mathbf{B}_{\ell}^{D}\mathbf{A}_{\ell}^{D} + \alpha\mathbf{B}_{\ell}^{P}\mathbf{A}_{\ell}^{P}\right]\mathbf{h}, \tag{7}$$

where $\mathbf{h}$ is the input feature to layer ℓ, and $(\mathbf{A}_{\ell}^{D}, \mathbf{B}_{\ell}^{D})$ and $(\mathbf{A}_{\ell}^{P}, \mathbf{B}_{\ell}^{P})$ are the low-rank factors of the D- and P-endpoint adapters, respectively. The resulting adapter output is added to the frozen layer output. Accordingly, the input and output convolution layers of the VAE decoder are also paired and interpolated by α in the same manner. The combined effect of generative-prior scaling and internal adapter interpolation is evaluated in Section 5.4.

4 How Training Objectives Shape D–P Control

This section examines how the objective used to train the codec constrains its attainable D–P control range and whether rate-adaptive D–P objective balancing alleviates these constraints. All controlled variants use the same architecture and are trained only at $\alpha \in \{0, 1\}$, excluding intermediate preferences to focus on endpoint performance.

4.1 Controlled Comparison of Objective-Specific Representations

We first train distortion-oriented and perception-oriented instances of the WIDER architecture. For this, we use two endpoint objectives, corresponding to the α=0 and α=1 settings of Equation 2, as

$$\mathcal{L}^{\alpha=0}(\mathbf{x}, q) = \mathcal{L}(\mathbf{x}, q, \alpha = 0) = \mathcal{L}_R(q) + \lambda_D(q)\mathcal{L}_D(\hat{\mathbf{x}}_q^0, \mathbf{x}), \tag{8}$$

$$\mathcal{L}^{\alpha=1}(\mathbf{x}, q) = \mathcal{L}(\mathbf{x}, q, \alpha = 1) = \mathcal{L}_R(q) + \lambda_P(q)\mathcal{L}_P(\hat{\mathbf{x}}_q^1, \mathbf{x}). \tag{9}$$

We refer to the compressed representations produced by the two trained codecs as distortion-oriented representation (DoR) and perception-oriented representation (PoR), respectively.

We then train only the decoder-side adaptation modules (*i.e.*, LoRA adapters and input and output convolution layers in U and V) for the opposite reconstruction objective. Specifically, such cross-objective training of the DoR variant minimizes $\mathcal{L}_P(\hat{\mathbf{x}}_q^1, \mathbf{x})$, whereas that for the PoR variant minimizes $\mathcal{L}_D(\hat{\mathbf{x}}_q^0, \mathbf{x})$. The compressed representation is preserved as in the training of the existing controllable codecs since the projections f and $\mathrm{LatentCodec}$ are frozen. The trainable parameters used by the two endpoints are disjoint. For example, the U-Net branch is active only at $\alpha = 1$ and is bypassed at $\alpha = 0$. Its weights are therefore not updated in the second phase, preserving the existing P-endpoint performance. Appendix A.3 specifies $\lambda_D(q)$ and $\lambda_P(q)$ for the DoR and PoR variants.

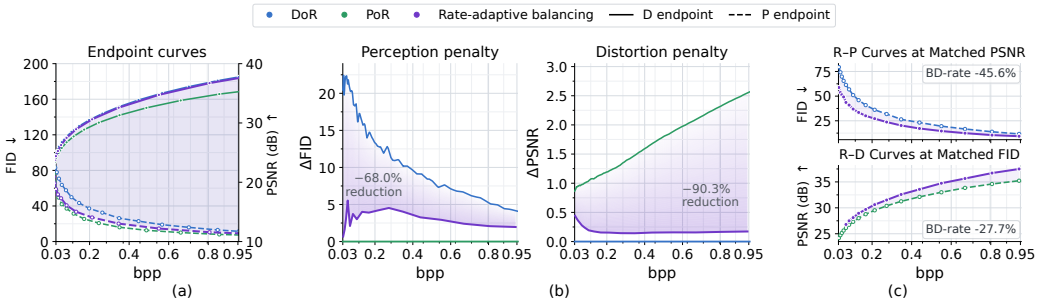


Figure 4: **Reduction of cross-endpoint penalties via rate-adaptive D–P objective balancing.** (a) D- and P-endpoint curves of the DoR variant, the PoR variant, and Rate-adaptive balancing (the endpoint-only WIDER variant); solid curves show D-endpoint PSNR on the right axis, whereas dashed curves show P-endpoint FID on the left axis. (b) The endpoint-only variant compared with the objective-specific variants at matched bpp. (c) Rate–perception and rate–distortion comparisons with the DoR and PoR variants after matching PSNR and FID, respectively.

4.2 COMPLEMENTARY CROSS-ENDPOINT PENALTIES

Figure 3 shows the quantitative results of this controlled experiment on Kodak24 and their visual examples. Despite using the same architecture and cross-objective decoder training procedure, the DoR and PoR variants attain clearly different D–P control ranges. The DoR variant retains higher PSNR at the D endpoint, whereas the PoR variant attains lower FID at the P endpoint. This separation shows that training a decoder for the opposite objective cannot fully overcome the endpoint-performance constraints imposed by the compressed representation. The penalties are asymmetric across rate: the perception penalty of DoR increases toward lower rates, whereas the distortion penalty of PoR increases toward higher rates. Notably, EGIC (Körber et al., 2024), the strongest baseline, follows the same training strategy as the DoR variant and achieves similar P-endpoint and D-endpoint performance in FID and PSNR, respectively. This implies that EGIC also holds a perception penalty from distortion-oriented training.

4.3 EXPANDING THE D–P CONTROL RANGE THROUGH RATE-ADAPTIVE BALANCING

We also construct an endpoint-only version of WIDER trained with our rate-adaptive D–P objective balancing in Section 3.2 only at the two endpoints, $\alpha \in \{0, 1\}$. Figure 4 labels this variant as Rate-adaptive balancing. Figure 4 shows how our method reduces the cross-endpoint penalties of the objective-specific representations. Specifically, it reduces the perception penalty of DoR relative to PoR by 68.0% and the distortion penalty of PoR relative to DoR by 90.3%, averaged across the evaluated rates. This result shows that a codec trained with rate-adaptive D–P objective balancing can approach both reference endpoint performances using a single shared representation.

The reduction of cross-endpoint penalty alone, however, does not reveal whether the endpoint-only variant merely forms an intermediate compromise between the DoR and PoR variants or provides a better RDP trade-off. The right panels therefore compare this variant with the DoR variant in rate–perception space after matching distortion, and compare it with the PoR variant in rate–distortion space after matching perception. The endpoint-only variant attains BD-rates of -45.6% and -27.7% in the rate–perception comparison with the DoR variant and the rate–distortion comparison with the PoR variant, respectively. These results demonstrate that rate-adaptive balancing does more than produce an intermediate point between the DoR and PoR variants: it makes the attainable D–P control range broader and improves the attainable RDP trade-off.

5 EXPERIMENTS

5.1 EXPERIMENTAL SETUP

Training. Training proceeds in three stages. *Stage 1*: a warm-up stage with high-rate R–D training without the diffusion U-Net for 457k steps . *Stage 2*: a main stage with the objective in Equation 2 for 270k steps , using only $\alpha \in \{0, 1\}$ and 64 discrete values of $q \in [0, 1]$ with uniform sampling. *Stage 3*: a fine-tuning stage with the discriminator loss for 30k steps . This stage samples α from eight uniformly spaced values according to a symmetric U-shaped distribution to learn intermediate points. Appendix A details the losses, optimization settings, and control parameter sampling.

Table 1: **Endpoint BD-rate of WIDER against each codec.** Negative value is better. Subscripts D/P denote D- and P-endpoint performances; †/* mark digitized/reimplemented results.

	Kodak24				CLIC 2020				DIV2K			
vs.	$\mathrm{PSNR_D}$	$\mathrm{FID_P}$	$\mathrm{LPIPS_P}$	$\mathrm{DISTS_P}$	$\mathrm{PSNR_D}$	$\mathrm{FID_P}$	$\mathrm{LPIPS_P}$	$\mathrm{DISTS_P}$	$\mathrm{PSNR_D}$	$\mathrm{FID_P}$	$\mathrm{LPIPS_P}$	$\mathrm{DISTS_P}$
EGIC	-2.8%	-50.3%	-39.6%	-51.0%	-3.1%	-60.0%	-49.6%	-57.6%	-5.6%	-51.7%	-44.8%	-55.0%
CRDR	-6.7%	-59.2%	-42.3%	-60.1%	-7.8%	-69.8%	-49.7%	-65.9%	-6.7%	-71.1%	-44.0%	-62.6%
MRIC*	-20.5%	-65.7%	-51.4%	-70.4%	-24.3%	-82.1%	-64.0%	-71.9%	-22.6%	-80.1%	-59.0%	-71.5%
CALICE	-2.5%	-80.1%	-62.4%	-84.9%	-4.7%	-96.6%	-78.4%	-89.5%	-0.7%	-93.5%	-67.4%	-86.5%
Zhou et al.†	-15.0%	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
DIRAC†	-3.3%	-80.7%	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A

Table 2: **Matched-quality RDP trade-offs of WIDER relative to EGIC.** At each rate, we match D-endpoint PSNR or P-endpoint FID and report complementary-metric BD-rate (negative is better). The PSNR values are matched within 0.01 dB of the target. N/A indicates no overlapping range.

Evaluation	Kodak24	CLIC 2020	DIV2K
BD-rate in terms of FID (w/ matched PSNR)	-70.6%	-58.4%	N/A
BD-rate in terms of PSNR (w/ matched FID)	-19.9%	-19.6%	-20.3%

Datasets. Training uses DIV2K train (Agustsson & Timofte, 2017), Flickr2K (Lim et al., 2017), CLIC 2020 Professional train/validation, CLIC 2021 test, and the first 10,000 LSDIR images, with random 512×512 crops and horizontal/vertical flips. LSDIR is omitted in *Stage 3*, following AEIC-ME (Zhang et al., 2026). Evaluation uses Kodak24, DIV2K validation, and CLIC 2020 test.

Evaluation protocol and metrics. We evaluate WIDER at 17 rates, $q = j/63$ for $j \in \{0, 4, \ldots, 60, 63\}$, and eight D–P operating points, $\alpha = k/7$ for $k \in \{0, \ldots, 7\}$. Endpoint curves are compared by BD-rate (Bjøntegaard, 2001) over their common quality range; negative values denote bitrate savings over the baseline. We measure distortion with PSNR and perceptual quality with FID. FID uses 256×256 patches with 128-pixel-overlapped grids. Although LPIPS and DISTS are not distributional divergences, we report them as supplementary measures of perceptual similarity.

Comparison methods. WIDER controls rate-distortion-perception with one weight set. We compare it with the RDP-controllable codecs CRDR (Iwai et al., 2024), Zhou et al. (Zhou et al., 2025), and CALICE (Spadaro et al., 2026). CRDR and CALICE are evaluated using their official implementations and checkpoints, whereas the results of Zhou et al. are digitized from the paper. For broader endpoint comparisons, we include the rate-specific D–P controllable codecs MRIC* (Agustsson et al., 2023), DIRAC (Ghouse et al., 2023), and EGIC (Körber et al., 2024). EGIC is evaluated using its official implementation and checkpoints. We also compare with diffusion-based generative codecs dedicated to ultra-low-rate settings: PerCo* (Careil et al., 2024), OSCAR (Guo et al., 2025), DiffEIC (Li et al., 2025b), OneDC (Xue et al., 2025), StableCodec (Zhang et al., 2025), and AEIC-ME (Zhang et al., 2026). * denotes results evaluated using publicly available reimplementations. Appendix K lists the repositories and control parameters used for the compared models.

5.2 MAIN RESULTS

Wide D–P control. Figure 1 shows that WIDER jointly provides strong D- and P-endpoint performance at the evaluated rates. Table 1 quantifies this result on Kodak24, CLIC 2020, and DIV2K. WIDER achieves substantial BD-rate gains against every competitor on their available endpoint metrics, including D–P controllable codecs optimized separately at each rate. Against the strongest baseline, EGIC, WIDER achieves BD-rate gains of up to -60.0%, -49.6%, and -57.6% in FID, LPIPS, and DISTS, respectively, and up to -5.6% in PSNR. These results show that the same compressed representation supports stronger performance at both ends of D–P control.

Better RDP trade-off. Endpoint BD-rate compares the best distortion and perception performance attained by each codec, but does not directly establish the RDP trade-off between the endpoints. Table 2 evaluates this property. At each rate, we adjust one codec's D–P balance to match the other codec's P-endpoint FID and then compare their rate–PSNR performance. Conversely, we match the other codec's D-endpoint PSNR and compare their rate–FID performance. We conduct this comparison against the strongest baseline, EGIC. WIDER achieves BD-rate gains of up to -20.3% in the FID-matched rate–PSNR comparison and -70.6% in the PSNR-matched rate–FID comparison. See Appendix B for details. These results show that WIDER does more than provide intermediate points between two strong endpoints: it improves the complementary objective at matched settings.

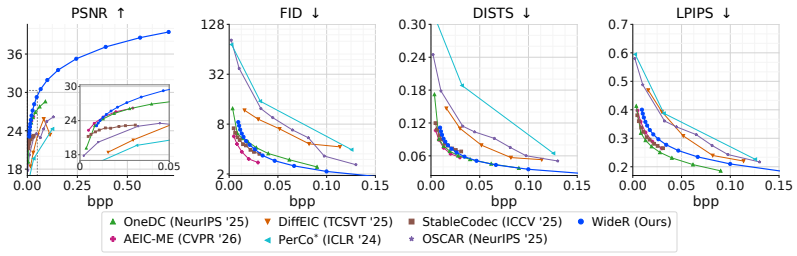


Figure 5: **Quantitative comparison with diffusion-based generative codecs on CLIC 2020.**



Figure 6: **Qualitative comparison with one-step diffusion-based codecs.** Full-image bpp and PSNR/LPIPS are reported for each reconstruction. At comparable ultra-low rates, the P endpoint of WIDER attains the highest PSNR while remaining visually comparable to the other methods.

Wide rate control. Figure 1 shows that WIDER supports D–P control from ultra-low-rate generative reconstruction to high-fidelity reconstruction. Its evaluated operating range extends beyond the available operating points of the compared controllable codecs, particularly toward ultra-low rates. Comparisons with dedicated generative codecs in Section 5.3 assess reconstruction quality in this extended region. Together, Figure 1, Table 1, and the matched-quality comparisons demonstrate the three design goals in Section 3.1: wide D–P control, a better RDP trade-off, and wide rate control.

5.3 COMPARISON WITH DIFFUSION-BASED GENERATIVE CODECS

To assess WIDER's performance as a diffusion-based codec, we also compare it with generative codecs targeting ultra-low rates (Figure 5). AEIC-ME and OneDC attain lower perceptual scores, whereas WIDER achieves higher PSNR, placing it at a different distortion–perception operating point. Qualitatively, WIDER produces visually comparable reconstructions while retaining source structure more faithfully (Figure 6). WIDER can thus serve as a baseline for future research toward wide-range controllable codecs that fully retain the reconstruction performance of models dedicated to ultra-low rates. Complete DIV2K, CLIC 2020, and Kodak24 curves are provided in Appendix H.

5.4 ABLATION STUDY

Rate-agnostic objective balancing. To assess the benefit of varying the D–P objective balance across rates, we train WIDER with a fixed-balance objective using $\lambda_D(q)/\lambda_P(q) = 4$ across rates. We use Equation 2 and preserve the rate-dependent overall coefficient scale of our endpoint-only variant to isolate this factor. Experiments show that the endpoint-only variant still achieves -18.1% R–D BD-rate at matched FID and -84.3% R–P BD-rate at matched PSNR against this fixed-balance variant, showing improved matched-quality rate efficiency over this fixed-balance control. Appendix C provides the full comparison.

Decoder-internal D–P control. We compare the decoder-internal D–P control described in Section 3.3 with linear pixel-space interpolation between the same endpoint reconstructions. Figure 7 reports eight rate-matched D–P operating points at three fixed rates on CLIC 2020. Decoder-internal control trained with an α sweep attains lower FID at comparable PSNR over most of each trajectory.

5.5 COMPLEXITY

Table 3 compares the parameter counts and coding latency of WIDER, AEIC-ME, OneDC, EGIC, and CRDR using their official implementations and native coding paths. On the same Kodak24 input,

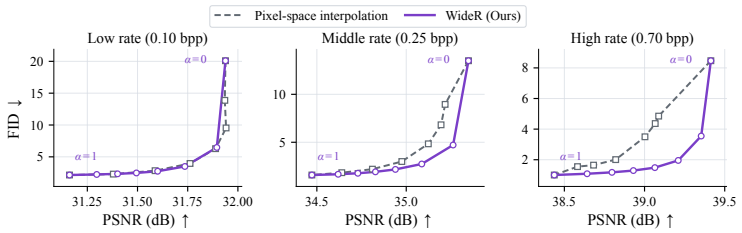


Figure 7: Absolute D–P traversal at fixed rates on CLIC 2020. Curves are interpolated along rate for visualization; markers correspond to the eight evaluated α values. Higher PSNR and lower FID are better.

Table 3: Comparisons of model complexity and coding latency.

				Parameters (M)			Latency (ms/image)	
Codec family	Model	Setting	Rate mode (# weight sets)	Total	Encoder	Decoder	Encoding	Decoding
One-step diffusion codec	AEIC-ME	P	Single (5)	926 M	56 M	922 M	18 ms	160 ms
	OneDC	P	Single (8)	1406 M	353 M	1058 M	117 ms	311 ms
D–P controllable codec	EGIC	D/P	Single (6)	33 M	23 M	20 M	120 ms	108 ms
RDP-controllable codec	CRDR	D/P	Variable (1)	128 M	114 M	116 M	784 ms	834 ms
	WideR (Ours)	$\alpha = 0$	Variable (1)	864 M	57 M	860 M	44 ms	98 ms
		$\alpha > 0$					44 ms	179 ms

we report the median of seven FP32 runs on an RTX 3090 with batch size one after two warm-ups. For WIDER, $\alpha = 0$ bypasses the diffusion U-Net, whereas $\alpha > 0$ activates it, accounting for the difference in decoding latency while using the same model parameters. Rate-specific methods require independently optimized weights for individual rates, so model storage grows with rate coverage. In contrast, WIDER amortizes one shared checkpoint over all rates and D–P balances. Hard-routing results (Section 5.4) support reducing U-Net computation as bitrate increases.

We also investigate whether U-Net computation can be reduced at higher rates, where the compressed representation retains more source information. We test hard routing that progressively omits deeper U-Net blocks as rate increases. Specifically, four, three, two, or one spatial levels are executed for quality indices $i_q \in [0, 31]$, $[32, 47]$, $[48, 55]$, and $[56, 63]$, respectively. Under the benchmarking conditions of Table 3, the three truncated routes take 71.13 ms ($7.8\% \downarrow$), 58.18 ms ($25.2\% \downarrow$), and 40.71 ms ($47.5\% \downarrow$) of U-Net execution time at $i_q = 32, 48, 56$, respectively, relative to full execution at each index. The routed model changes R–D and R–P BD-rate by only $+0.10\%$ and $+1.35\%$ relative to the full model. The details of implementation and optimization are described in Appendix E.

6 LIMITATIONS

The four boundary weights in Section 3.2 are chosen empirically rather than systematically optimized. Comparing alternative settings is complicated because they affect both the D–P balance and rate, while their numerical ratios may not reflect the relative influence of the two objectives during codec optimization. Isolating their effects therefore requires jointly retuning the overall loss scale and rate–distortion multiplier, beyond a single-factor ablation. Nevertheless, our results show that the reconstruction objective used to train the codec constrains its attainable D–P control range, and that a simple geometric schedule substantially alleviates the resulting cross-endpoint penalties, motivating better scheduling or balancing-parameter optimization.

7 CONCLUSION

The cross-endpoint penalties revealed in this work suggest that what information a controllable codec retains in its shared compressed representation should be a key design consideration for wide D–P control. We present one approach to this problem and show that even a simple weighting schedule substantially alleviates these penalties and improves endpoint performance. Future work could explore better balancing methods or optimal hyperparameters, as discussed in Section 6. We expect WIDER to serve as a useful baseline bridging controllable compression and ultra-low-rate generative compression through D–P control.

AI USE STATEMENT

We used a generative AI assistant in preparing this work. The table groups activities according to the ICLR 2027 AI Policy for Authors. The scope below reflects the project record and author confirmation.

Task with required disclosure	Used	Scope
Generate synthetic data sets	No	
Help develop theoretical models or conceptual frameworks	No	
Formulate mathematical claims	No	
Provide critical ingredients for proving mathematical claims	No	
Assist in the writing of proofs	No	
Propose or refine hypotheses	No	
Design or provide feedback on research methodology or experiments	No	
Implement methods	No	The method was designed by the authors; AI assistance was limited to code edits reported below.
Assist with translation	Yes	Translation support for manuscript revisions specified by the authors.
Clean and reformat dataset	N/A	
Support qualitative and thematic data analysis	N/A	
Interpret results	No	
Task with recommended disclosure	**Used**	**Scope**
Formulate questions for surveys or interviews	No / N/A	
Create or modify scientific figures or images	Yes	Plotting code for Figures 4 and 15–20, under the authors' specifications.
Suggest experimental parameters	No	
Create or edit software code	Yes	Edits to training, evaluation, analysis, and plotting code under the authors' direction.
Creation of artifacts	No	
Draft parts of a research paper	Yes	Assistance with drafting appendix sections.
Transcribe recordings of research material	No / N/A	
Summarize or analyse existing literature	No	
Discover research topics or identify gaps	No	
Brainstorming	Yes	Discussion of presentation options with the authors.
Sourcing/searching for information	Yes	Searches for recent controllable-compression work and verification of published settings.
Edit a research paper to improve readability	Yes	Condensing and copy-editing.
Identify relevant literature	Yes	Identification of recent relevant work, including CALICE (2026).
Format references	Yes	BibTeX formatting assistance.
Suggest a structure for a research paper	No	
Propose a title or keywords for a research paper	No	

We have reviewed all AI-assisted work: all code was reviewed and tested by the authors, every reported number was produced by the described pipelines on the stated data, and all AI-drafted text was checked against the experiments and cited sources. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work uses established public image-compression benchmarks and pretrained models; it introduces no new dataset, personal-data collection, or study with recruited participants. As with other generative codecs, perception-oriented reconstructions may contain plausible details absent from the source and should be used with care in applications requiring faithful visual records.

Reproducibility statement

For components adopted unchanged from prior work, we refer readers to the original publications. Details of our modifications and experimental procedures are provided in the main text and appendix. We plan to release code and trained checkpoints publicly upon publication.

References

Eirikur Agustsson and Radu Timofte. NTIRE 2017 challenge on single image super-resolution: Dataset and study. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 126–135, 2017.

Eirikur Agustsson, David Minnen, George Toderici, and Fabian Mentzer. Multi-realism image compression with a conditional generator. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22324–22333, 2023.

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

Jean Bégaint, Fabien Racapé, Simon Feltman, and Akshay Pushparaja. CompressAI: A PyTorch library and evaluation platform for end-to-end compression research. *arXiv preprint arXiv:2011.03029*, 2020.

Gisle Bjøntegaard. Calculation of average PSNR differences between RD-curves. ITU-T SG16/Q6, Video Coding Experts Group, VCEG-M33, 2001.

Yochai Blau and Tomer Michaeli. Rethinking lossy compression: The rate-distortion-perception tradeoff. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 675–685, 2019.

Marlene Careil, Matthew J. Muckley, Jakob Verbeek, and Stéphane Lathuilière. Towards image compression with perfect realism at ultra-low bitrates. In *International Conference on Learning Representations*, 2024.

Bin Chen, Gehui Li, Rongyuan Wu, Xindong Zhang, Jie Chen, Jian Zhang, and Lei Zhang. Adversarial diffusion compression for real-world image super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 28208–28220, 2025.

Jierun Chen, Shiu-hong Kao, Hao He, Weipeng Zhuo, Song Wen, Chul-Ho Lee, and S.-H. Gary Chan. Run, don't walk: Chasing higher FLOPS for faster neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12021–12031, 2023.

Keyan Ding, Kede Ma, Shiqi Wang, and Eero P. Simoncelli. Image quality assessment: Unifying structure and texture similarity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(5):2567–2581, 2022. doi: 10.1109/TPAMI.2020.3045810.

Noor Fathima Ghouse, Jens Petersen, Auke Wiggers, Tianlin Xu, and Guillaume Sautière. Neural image compression with a diffusion-based decoder, 2023. URL `https://arxiv.org/abs/2301.05489v2`. Version 2, January 23, 2023.

Jinpei Guo, Yifei Ji, Zheng Chen, Kai Liu, Min Liu, Wang Rao, Wenbo Li, Yong Guo, and Yulun Zhang. OSCAR: One-step diffusion codec across multiple bit-rates. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-2857.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL `https://openreview.net/forum?id=nZeVKeeFYf9`.

Shoma Iwai, Tomo Miyazaki, and Shinichiro Omachi. Controlling rate, distortion, and realism: Towards a single comprehensive neural image compression model. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 2900–2909, 2024.

Nikolai Körber, Eduard Kromer, Andreas Siebert, Sascha Hauke, Daniel Mueller-Gritschneder, and Björn Schuller. EGIC: Enhanced low-bitrate generative image compression guided by semantic segmentation. In *European Conference on Computer Vision*, volume 15093 of *Lecture Notes in Computer Science*, pp. 202–220. Springer, 2024. doi: 10.1007/978-3-031-72761-0_12.

Jiahao Li, Bin Li, and Yan Lu. Neural video compression with diverse contexts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22616–22626, 2023.

Jiahao Li, Bin Li, and Yan Lu. Neural video compression with feature modulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 26099–26108, 2024.

Yuqi Li, Haotian Zhang, Li Li, and Dong Liu. Learned image compression with hierarchical progressive context modeling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 18834–18843, 2025a.

Zhiyuan Li, Yanhui Zhou, Hao Wei, Chenyang Ge, and Jingwen Jiang. Toward extreme image compression with latent feature guidance and diffusion prior. *IEEE Transactions on Circuits and Systems for Video Technology*, 35(1):888–899, 2025b. doi: 10.1109/TCSVT.2024.3455576.

Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 136–144, 2017.

Xu Ma, Xiyang Dai, Yue Bai, Yizhou Wang, and Yun Fu. Rewrite the stars. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5694–5703, 2024.

Daniele Mari and Simone Milani. Enhancing the rate-distortion-perception flexibility of learned image codecs with conditional diffusion decoders. arXiv:2403.02887, 2024. URL `https://arxiv.org/abs/2403.02887`.

Fabian Mentzer, George Toderici, Michael Tschannen, and Eirikur Agustsson. High-fidelity generative image compression. In *Advances in Neural Information Processing Systems*, volume 33, pp. 11913–11924, 2020.

Matthew J. Muckley, Alaaeldin El-Nouby, Karen Ullrich, Hervé Jégou, and Jakob Verbeek. Improving statistical fidelity for neural image compression with implicit local likelihood models. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 25426–25443, 2023.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763, 2021.

Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation. In *European Conference on Computer Vision*, volume 15144 of *Lecture Notes in Computer Science*, pp. 87–103. Springer, 2024. doi: 10.1007/978-3-031-73016-0_6.

Gabriele Spadaro, Alberto Presta, Jhony H. Giraldo, Attilio Fiandrotti, Marco Grangetto, and Enzo Tartaglione. CALICE: Continuous bitrate control with adapted LIC model. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 2026. doi: 10.1145/3820658. URL `https://doi.org/10.1145/3820658`.

Naifu Xue, Zhaoyang Jia, Jiahao Li, Bin Li, Yuan Zhang, and Yan Lu. One-step diffusion-based image compression with semantic distillation. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-1247.

Zeyu Yan, Fei Wen, Rendong Ying, Chao Ma, and Peilin Liu. On perceptual lossy compression: The cost of perceptual reconstruction and an optimal training framework. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 11682–11692, 2021.

Zeyu Yan, Fei Wen, and Peilin Liu. Optimally controllable perceptual lossy compression. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 24911–24928, 2022.

Weihao Yu and Xinchao Wang. MambaOut: Do we really need mamba for vision? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4484–4496, 2025.

Weihao Yu, Pan Zhou, Shuicheng Yan, and Xinchao Wang. InceptionNeXt: When inception meets ConvNeXt. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5672–5683, 2024.

George Zhang, Jingjing Qian, Jun Chen, and Ashish Khisti. Universal rate-distortion-perception representations for lossy compression. In *Advances in Neural Information Processing Systems*, volume 34, pp. 11517–11529, 2021.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 586–595, 2018.

Tianyu Zhang, Xin Luo, Li Li, and Dong Liu. StableCodec: Taming one-step diffusion for extreme image compression. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 17379–17389, 2025.

Tianyu Zhang, Dong Liu, and Chang Wen Chen. Ultra-low bitrate perceptual image compression with shallow encoder. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12118–12128, 2026.

Chuqin Zhou, Guo Lu, Jiangchuan Li, Xiangyu Chen, Zhengxue Cheng, Li Song, and Wenjun Zhang. Controllable distortion-perception tradeoff through latent diffusion for neural image compression. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 10725–10733, 2025. doi: 10.1609/aaai.v39i10.33165.

A Details of Model Training

A.1 Sampling of q and α

We discretize the normalized rate-control parameter as $q = j/63$ for $j \in \{0, \ldots, 63\}$. The 64 indices are assigned by a deterministic shuffled cycle without replacement, ensuring uniform coverage.

For the final WideR model, the treatment of α changes across training phases as summarized in Table 4. During the D–P balance sweep, $\alpha = k/7$ for $k \in \{0, \ldots, 7\}$ is sampled according to

$$\Pr(\alpha = k/7) = 0.9 \frac{[1 + \min(k, 7-k)]^{-1/2}}{\sum_{i=0}^{7} [1 + \min(i, 7-i)]^{-1/2}} + 0.05\, \mathbf{1}_{\{0,7\}}(k). \tag{10}$$

This symmetric distribution has mean $1/2$ and allocates additional mass to the two endpoints while retaining all six interior D–P balances. During the balance sweep, the q and α draws are independent. For the controlled comparisons, endpoint-only WideR is trained on the paired endpoints $\alpha \in \{0, 1\}$, whereas the DoR and PoR variants use only $\alpha = 0$ and 1, respectively.

A.2 Implementation of $\mathcal{L}_P$

Following Equation 2, all reconstruction losses take the reconstruction first and the source image second. We implement the distortion loss as

$$\mathcal{L}_D(\hat{\mathbf{x}}_q^\alpha, \mathbf{x}) = \mathcal{L}_{\mathrm{MSE}}(\hat{\mathbf{x}}_q^\alpha, \mathbf{x}). \tag{11}$$

The perception loss $\mathcal{L}_P(\hat{\mathbf{x}}_q^\alpha, \mathbf{x})$ uses LPIPS–VGG (Zhang et al., 2018), DISTS (Ding et al., 2022), CLIP feature distance (Radford et al., 2021), and an adversarial loss. We instantiate it with the following two phase-specific forms:

$$\text{Stage 2:} \quad \mathcal{L}_P(\hat{\mathbf{x}}, \mathbf{x}) = 2\mathcal{L}_{\mathrm{MSE}}(\hat{\mathbf{x}}, \mathbf{x}) + \mathcal{L}_{\mathrm{LPIPS}}(\hat{\mathbf{x}}, \mathbf{x}) + 0.1\mathcal{L}_{\mathrm{CLIP}}(\hat{\mathbf{x}}, \mathbf{x}), \tag{12}$$

$$\text{Stage 3:} \quad \mathcal{L}_P(\hat{\mathbf{x}}, \mathbf{x}) = 2\mathcal{L}_{\mathrm{MSE}}(\hat{\mathbf{x}}, \mathbf{x}) + \mathcal{L}_{\mathrm{DISTS}}(\hat{\mathbf{x}}, \mathbf{x}) + 0.1\mathcal{L}_{\mathrm{CLIP}}(\hat{\mathbf{x}}, \mathbf{x}) + 0.1\mathcal{L}_{\mathrm{GAN}}(\hat{\mathbf{x}}). \tag{13}$$

Stage 2 uses the non-adversarial form during codec training and cross-objective decoder learning for the P endpoint, whereas Stage 3 uses the adversarial form during refinement. At the two endpoints, Equation 2 evaluates $\mathcal{L}_D(\hat{\mathbf{x}}_q^0, \mathbf{x})$ and $\mathcal{L}_P(\hat{\mathbf{x}}_q^1, \mathbf{x})$, respectively. For the DoR and PoR variants, the same functions are evaluated on their corresponding endpoint reconstructions $\hat{\mathbf{x}}_q^0$ and $\hat{\mathbf{x}}_q^1$.

In Stage 3, $\mathcal{L}_{\mathrm{DISTS}}$ is implemented as overlap-chunked edge-aware DISTS (OC-EA-DISTS), which evaluates DISTS on overlapping image patches and their Sobel edge maps with weights 1.5 and 0.5, respectively. The GAN term uses the generator loss from a frozen DINOv2 feature backbone and a trainable discriminator head.

We use the following public implementations:

- OC-EA-DISTS: `https://github.com/LuizScarlet/AEIC`
- LPIPS: `https://github.com/richzhang/PerceptualSimilarity`
- DISTS: `https://github.com/chaofengc/IQA-PyTorch`
- CLIP: `https://github.com/openai/CLIP`
- DINOv2: `https://github.com/facebookresearch/dinov2`

A.3 Stage-wise Training Details

Section 5.1 summarizes the three stages used to train the final WideR model. This section gives the objective schedules, optimization settings, and trainable components for those stages, followed by the corresponding protocol for the DoR, PoR, and endpoint-only WideR variants in Section 4.

Table 4: Three-stage training curriculum for the final WideR model.

Stage	Objective	q	α	Updates	Batch	Learning rate
1	$\mathcal{L}_R + 1024\mathcal{L}_D$	1	0	457k	8	1e−4
2	Eq. 2; $\mathcal{L}_P$: Eq. 12	$\{j/63\}_{j=0}^{63}$	$\{0, 1\}$	270k	8	1e−4
3	Eq. 2; $\mathcal{L}_P$: Eq. 13	$\{j/63\}_{j=0}^{63}$	$\{k/7\}_{k=0}^{7}$	20k/5k/5k	32	5e−5/1e−5/1e−6

Table 5: Four-stage training curricula for the DoR and PoR variants. The projection f and latent codec are frozen before the opposite endpoint decoder is trained.

Variant	Stage	Objective	α	Updates	Batch	Learning rate
DoR/PoR	1	$\mathcal{L}_R + 1024\mathcal{L}_D$	0	457k	8	$1e{-4}$
DoR	2	Eq. 8	0	300k	8	$1e{-4}$ (275k), $5e{-5}$ (15k), $1e{-5}$ (5k), $1e{-6}$ (5k)
	3	Eq. 12	1	120k	8	$1e{-4}$
	4	Eq. 13	1	20k/5k/5k	32	$5e{-5}/1e{-5}/1e{-6}$
PoR	2	Eq. 9; $\mathcal{L}_P$: Eq. 12	1	270k	8	$1e{-4}$
	3	Eq. 9; $\mathcal{L}_P$: Eq. 13	1	20k/5k/5k	32	$5e{-5}/1e{-5}/1e{-6}$
	4	Eq. 11	0	130k/10k/5k/5k	8	$1e{-4}/5e{-5}/$ $1e{-5}/1e{-6}$

Training data and objective schedules. The full training set contains 14,106 images from DIV2K, Flickr2K, CLIC 2020 Professional, and the first 10K LSDIR images. Adversarial phases use the 4,046-image AEIC-ME training set, of which 4,029 images admit a 512×512 crop. This set includes CLIC 2021 test images. All runs use random 512×512 crops with horizontal and vertical flips. Appendix A.1 specifies the q and α sampling rules.

The distortion weight used for the DoR variant and the perception weight used for the PoR variant are, respectively,

$$\lambda_D(q) = 2^{10q}, \qquad \lambda_P(q) = 2^{-5+10q}, \tag{14}$$

whereas the endpoint-only and final WIDER models use

$$\lambda_D(q) = 2^{-9+20q}, \qquad \lambda_P(q) = 2^{-4+6q}. \tag{15}$$

Schedule-specific details. The final WIDER model and all controlled variants inherit the common Stage 1 checkpoint in Tables 4 and 5. The diffusion U-Net is bypassed during this warm-up, which is excluded from the post-warm-up training budget. At the start of post-warm-up training, the quality-index gains are initialized to identity and the endpoint modules use deterministic initialization; only the four gain tensors are updated for the first 10k steps. During codec training, the DoR and PoR variants optimize the projection f and the latent codec together with the adaptation modules for the D and P endpoints, respectively, whereas the endpoint-only model follows the Stage 2–3 schedule of WIDER while restricting α to the two endpoints, and optimizes these shared modules with both endpoint objectives. The DoR and PoR variants then freeze f and the latent codec and train only the adaptation modules for the opposite endpoint, as summarized in Table 5. No decoder-side quality-index parameters are introduced; the U-Net and VAE decoder receive rate information through $\hat{\mathbf{z}}_q$. The implementation absorbs the normalization factor into $\lambda_D(q)$ and $\lambda_P(q)$, giving effective D–P balancing multipliers $(1-\alpha)\lambda_D(q)$ and $\alpha\lambda_P(q)$.

After Stage 1, the total update counts are 450k for the DoR and PoR variants and 300k for the endpoint-only and final WIDER models. The controlled comparison holds the 300k codec-training budget before any opposite-endpoint adaptation fixed; the additional 150k updates for the DoR and PoR variants affect only their opposite endpoint decoders.

Optimization. We optimize the trainable generator parameters with AdamW, using a weight decay of 0.01 and gradient-norm clipping at 1.0, and reinitialize the optimizer at each phase boundary. During adversarial refinement, the frozen DINOv2 backbone is paired with a separately initialized trainable head optimized at a learning rate of 2×10^{-5} and discarded at evaluation.

B MATCHED-QUALITY COMPARISON WITH EGIC

Figure 8 shows the matched-quality curves summarized in Table 2.

Each dataset has six EGIC rate settings, evaluated with EGIC's preference control set to 0 for the distortion endpoint and 1 for the perception endpoint.

Table 6 lists their reference values in increasing bitrate order.

For PSNR matching, we search WIDER's α at each evaluated q until the dataset-mean PSNR differs from the distortion-endpoint target by at most 0.01 dB.

Only EGIC's reference curve is interpolated along rate to define this target at WIDER's measured bitrate; both WIDER metrics are computed from the same full-dataset reconstructions.

The resulting path uses four evaluated quality indices per dataset, listed with their searched α and measured PSNR errors in Table 7.

These rates generally differ from EGIC's six reference rates.

For FID matching, we interpolate measured metrics along log bitrate using PCHIP and linearly between eight sampled preferences, $\alpha \in \{0, 1/7, \ldots, 1\}$, to match EGIC's perception-endpoint FID, then compare PSNR.

Table 6 reports the resulting preference coordinates α_{interp} at EGIC's reference rates within the common rate range.

These coordinates are derived from the interpolated metric surface; reconstruction-based FID errors at these coordinates have not been measured and are marked with a dash.

An exact crossing on the interpolated surface does not establish an exact FID match after decoding.

BD-rates integrate the resulting paths over their overlapping quality ranges without extrapolation.

On DIV2K, the PSNR-matched paths have no overlapping FID range, so the corresponding BD-rate is reported as N/A.

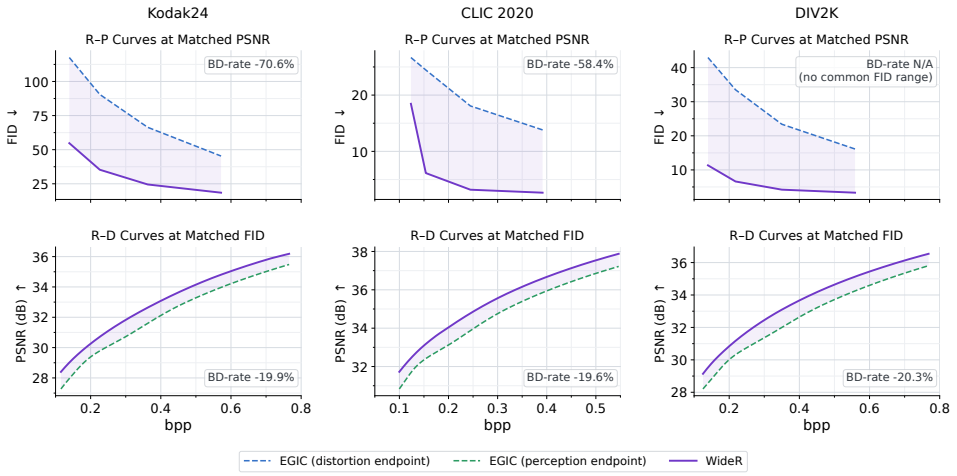


Figure 8: Matched-quality RDP comparisons with EGIC. Columns show Kodak24, CLIC 2020, and DIV2K. The upper row compares FID at matched PSNR using actual reconstructions; the lower row compares PSNR at matched FID using metric interpolation. Dashed curves show the corresponding EGIC endpoints, and solid curves show WIDER.

Table 6: EGIC reference points and FID-matching preference coordinates. PSNR and FID are the distortion- and perception-endpoint reference values, respectively. α_{interp} is derived by metric interpolation. Dashes in the last column denote unmeasured reconstruction-based FID errors; the missing α_{interp} at the lowest CLIC 2020 rate indicates that it lies outside the sampled WIDER rate range.

Dataset	Setting	bpp	PSNR (dB)	FID	α_{interp}	Measured $\lvert\Delta\mathrm{FID}\rvert$
Kodak24	1	0.1153	28.44	56.29	0.1966	—
	2	0.1405	29.09	50.19	0.2163	—
	3	0.1933	30.16	41.02	0.2461	—
	4	0.3011	31.77	31.25	0.2731	—
	5	0.4669	33.76	23.19	0.2794	—
	6	0.7658	36.39	13.27	0.5350	—
CLIC 2020	1	0.0854	31.45	3.79	—	—
	2	0.1019	32.03	3.42	0.2957	—
	3	0.1367	33.05	2.94	0.3454	—
	4	0.2064	34.45	2.13	0.5104	—
	5	0.3213	36.11	1.84	0.5294	—
	6	0.5465	38.32	1.42	0.6007	—
DIV2K	1	0.1253	29.24	9.00	0.2737	—
	2	0.1515	29.89	8.08	0.2849	—
	3	0.2026	30.95	6.48	0.3600	—
	4	0.3055	32.48	5.01	0.4164	—
	5	0.4681	34.30	3.81	0.5120	—
	6	0.7683	36.70	2.61	0.6799	—

Table 7: Control parameters and measured errors for actual PSNR matching. The integer quality index divided by 63 gives normalized q. Target PSNR is obtained from EGIC's distortion-endpoint reference curve at the measured bitrate. $|\Delta\mathrm{PSNR}|$ is the absolute difference between the measured and target values. PSNRs and errors are rounded to two decimals; all unrounded errors are below 0.01 dB.

Dataset	Quality index	bpp	α	Target PSNR (dB)	Measured PSNR (dB)	$\lvert\Delta\mathrm{PSNR}\rvert$ (dB)
Kodak24	44	0.1392	0.1636	29.06	29.06	0.00
	48	0.2261	0.2744	30.70	30.70	0.00
	52	0.3627	0.3923	32.56	32.56	0.00
	56	0.5718	0.3331	34.79	34.79	0.00
CLIC 2020	46	0.1235	0.0000	32.70	32.69	0.01
	48	0.1538	0.1291	33.45	33.45	0.00
	52	0.2446	0.2326	35.07	35.07	0.00
	56	0.3916	0.2403	36.91	36.91	0.00
DIV2K	44	0.1395	0.1856	29.60	29.61	0.00
	48	0.2180	0.3165	31.21	31.21	0.00
	52	0.3500	0.5410	33.03	33.03	0.00
	56	0.5588	0.5758	35.12	35.12	0.00

C Details of the Fixed-Balance Representation

For the fixed-balance representation (FBR) evaluated in Section 5.4, we use a positive, finite objective ratio that is constant across rates:

$$\frac{\lambda_D(q)}{\lambda_P(q)} = 4, \qquad \lambda_D(q)\lambda_P(q) = 2^{-13+26q}, \tag{16}$$

where $\lambda_D(q)$ and $\lambda_P(q)$ denote the weights used for FBR, and the second condition matches the product of the rate-adaptive weights in Equation 15 at every q. The fixed ratio 4 is the geometric mean of the rate-adaptive schedule's $\lambda_D(q)/\lambda_P(q)$ under the uniform quality-index distribution, not a separately tuned value. Substituting the rate-adaptive schedule in Equation 15 gives

$$\lambda_D(q) = 2^{-5.5+13q}, \qquad \lambda_P(q) = 2^{-7.5+13q}. \tag{17}$$

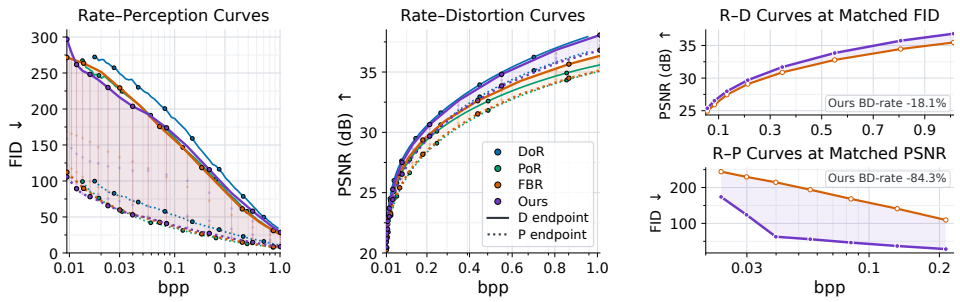


Figure 9: Rate-adaptive versus fixed D–P objective balancing on Kodak24. The left and center panels compare the DoR variant, the PoR variant, FBR, and endpoint-only WIDER (Ours). Solid/dotted curves denote D/P endpoints; shaded bands and faint interior points show the measured D–P control ranges of FBR and Ours. At right, FBR is matched to the rate-adaptive model's P-endpoint FID for the R–D comparison (top), whereas the rate-adaptive model is matched to FBR's D-endpoint PSNR for the R–P comparison (bottom). BD-rates use FBR as the anchor over each matched measured range. No extrapolation is used.

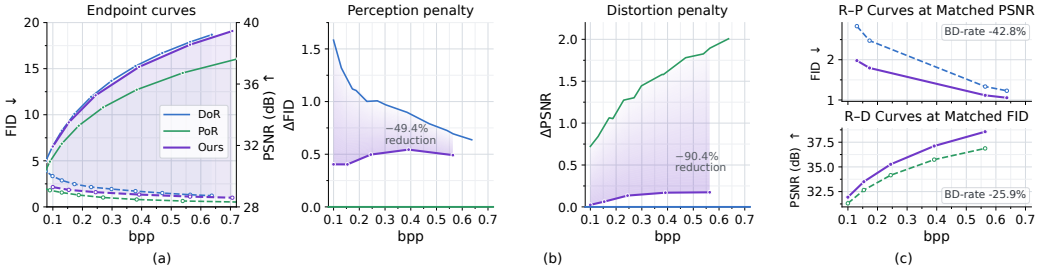


Figure 10: Reduction of cross-endpoint penalties on CLIC 2020. We repeat the endpoint, matched-bpp penalty, and matched-quality comparisons of Figure 4 on all 428 test images. For PSNR matching, endpoint mixtures of the DoR variant are evaluated at four actual q values within 0.01 dB of the target; FID uses the same reconstructions. Only feasible matches within measured rate support are included.

Initialization, trainable parameters, endpoint decoders, data and sampling policies, effective batch size, optimization phases, and refinement losses are matched to endpoint-only WIDER, which serves as the rate-adaptive reference. Thus, FBR and endpoint-only WIDER optimize the same objective in Equation 2 and differ only in whether $\lambda_D(q)/\lambda_P(q)$ is constant or varies with q.

We compare FBR and endpoint-only WIDER through their endpoint curves and two complementary matched-quality paths. The left and center panels of Figure 9 also include the DoR and PoR variants as objective-specific references, placing fixed and rate-adaptive balancing alongside the cross-endpoint limitations studied in Section 4. For the matched R–D path, endpoint-only WIDER's P-endpoint FID is the target and FBR's eight-point α sweep is interpolated to that FID. For the matched R–P path, FBR's D-endpoint PSNR is the target and the endpoint-only WIDER variant is decoded at a searched α until dataset-mean PSNR matches within 0.01 dB. FID is then measured from those same reconstructions. We do not extrapolate beyond either measured D–P sweep.

D ENDPOINT-PENALTY REDUCTION ON CLIC 2020

To verify that the findings on Kodak24 are consistent across datasets, we repeat the endpoint-penalty analysis in Figure 4 on all 428 images of the CLIC 2020 test set. Figure 10 shows the same trend: endpoint-only WIDER reduces the perception penalty of DoR by 49.4% and the distortion penalty of PoR by 90.4% on average. At matched PSNR and FID, it further attains BD-rates of -42.8% relative to the DoR variant and -25.9% relative to the PoR variant, respectively. Thus, the cross-endpoint penalty reductions observed on Kodak24 remain consistent on the larger and higher-resolution CLIC 2020 test set.

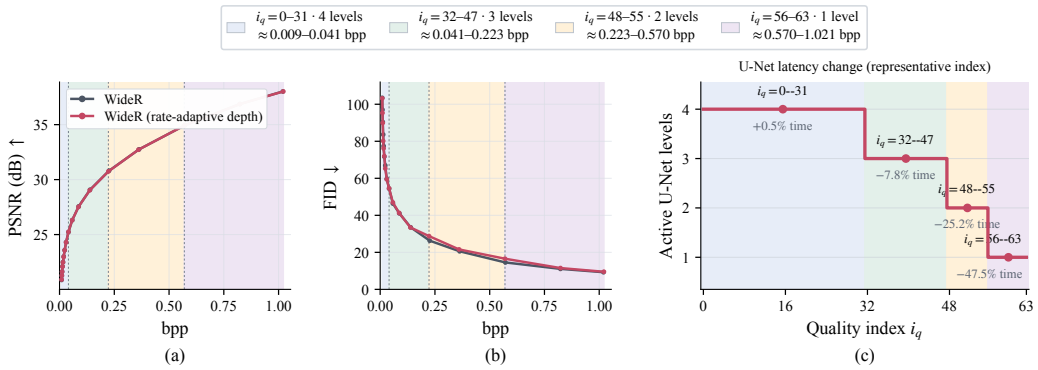


Figure 11: Rate-adaptive U-Net depth on Kodak24. (a,b) R–D and R–P endpoint curves of WIDER and the routed variant. Colors indicate routing intervals in approximate Kodak24 mean bpp, with boundaries at the routed model's measured group-start rates. (c) The four quality-index intervals and their executed spatial levels; U-Net latency changes are measured at representative indices $i_q = 0, 32, 48, 56$ relative to full execution at each index under the conditions of Table 3.

E RATE-ADAPTIVE U-NET DEPTH

Problem. The one-step diffusion prior normally incurs the same four-level U-Net cost for generative reconstruction at all rates. This is inefficient at high rates: the compressed representation already carries more source evidence, yet the decoder still executes the deepest generative path used at ultra-low rates. We test whether U-Net capacity can instead be allocated according to the rate without materially narrowing the endpoint performance range.

Hard depth routing. Let $i_q \in \{0, \dots, 63\}$ be the discrete quality index corresponding to the normalized rate-control parameter q. The number of active U-Net spatial levels is

$$d(i_q) = \begin{cases} 4, & 0 \le i_q \le 31, \\ 3, & 32 \le i_q \le 47, \\ 2, & 48 \le i_q \le 55, \\ 1, & 56 \le i_q \le 63. \end{cases} \tag{18}$$

Figure 12 illustrates the executed blocks. This is hard execution routing rather than feature masking. For a truncated route, the omitted deeper down blocks, bottleneck, and mirrored up blocks are not executed; the first retained up block starts from a zero trunk and consumes the retained skip features. Thus, the mechanism adds no parameters and leaves the U-Net unchanged at $i_q \le 31$. We reuse the same Stage 2 checkpoint as WIDER and train the routed variant only during the final 30k-update fine-tuning stage (Stage 3), without repeating Stages 1 and 2. The fine-tuning protocol is unchanged apart from the routing rule.

U-Net latency by routing interval. We use the same Kodak24 image (`kodim01`, 768×512), RTX 3090, FP32 precision, batch size one, and optimized decoding implementation as Table 3, with $\alpha = 0.5$. At representative indices $i_q = 0, 32, 48, 56$, we measure full and routed models on the same GPU for each pair. U-Net calls are timed separately within full decoding runs using synchronized wall-clock timing, reporting the median of seven runs after two warm-ups. Full U-Net execution takes 77.76, 77.14, 77.81, and 77.58 ms, respectively; the four routed paths take 78.15, 71.13, 58.18, and 40.71 ms. Thus, the $3/2/1$-level routes reduce U-Net latency by 7.8%, 25.2%, and 47.5%, respectively. The untruncated four-level route is 0.5% slower in this measurement. These are representative-index measurements, not an exhaustive average over all 64 indices, and the reductions refer to U-Net execution rather than total decoding latency. As shown in Figure 11, this saving is obtained with only $+0.10\%$ R–D and $+1.35\%$ R–P BD-rate relative to WIDER.

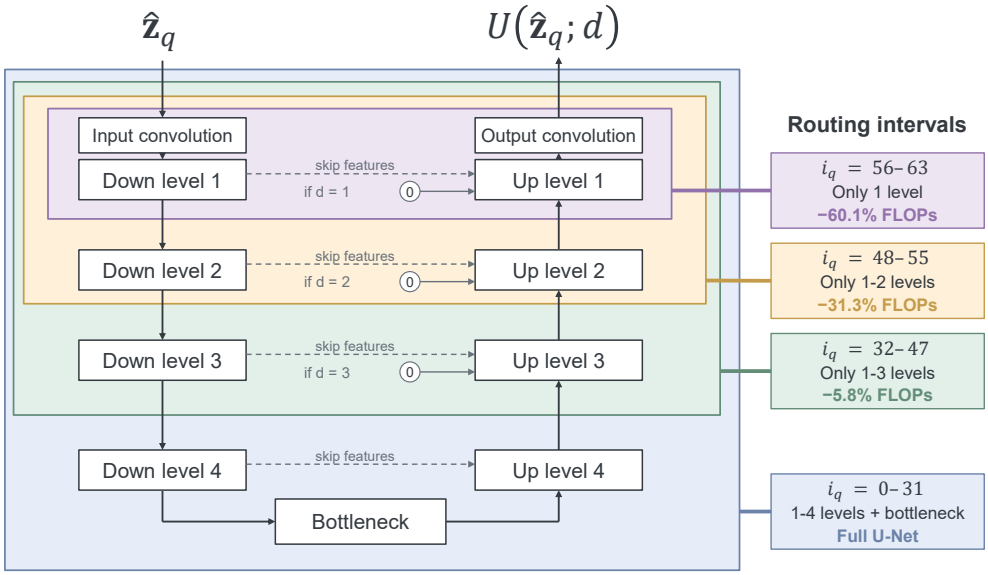


Figure 12: Rate-conditioned hard routing in the U-Net. Nested range boxes indicate the spatial levels retained by each route; colors match Figure 11. The full route includes the bottleneck. Truncated routes omit deeper blocks and the bottleneck, skip downsampling at the deepest retained level, and initialize the first retained up block with a zero trunk while preserving its skip inputs. Input and output convolutions are shared across all routes. Individual residual and attention blocks and multiple skip features are grouped for clarity.

F Model Architecture

This appendix specifies the controllable-codec construction in Section 3.3. The description follows the rate-adaptive model used in our final evaluation. The objective-specific representations use the same input, compressed-representation, and endpoint interfaces; their different optimization procedures are given in Section 4.1.

F.1 Pixel Projection and Variable-Rate Latent Codec

The pixel projection f uses three downsampling stages to produce the 256-channel reconstruction latent $\mathbf{z} \in \mathbb{R}^{H/8 \times W/8 \times 256}$. LatentCodec applies the analysis transform g_a to obtain a 320-channel coded latent and the synthesis transform g_s to recover $\hat{\mathbf{z}}_q$ in the 256-channel reconstruction space (Equation 5). Thus, g_s terminates in latent space and leaves pixel reconstruction to the endpoint decoding functions. Table 8 specifies the three transforms. They use the partial-convolution residual blocks of HPCM-Base (Li et al., 2025a), whose spatial mixing originates from FasterNet (Chen et al., 2023).

In Table 8, N denotes the number of output channels, $\downarrow 2$ and $\uparrow 2$ denote twofold spatial resampling, and $\times n$ denotes n consecutive blocks. Unmarked convolutions use stride one. The stride-two 4×4 convolution uses padding one, whereas stride-two 2×2 convolutions use no padding. For the 64 supported rate settings, $i_q \in \{0, \ldots, 63\}$ is the discrete quality index corresponding to $q = i_q/63$. $M_E[i_q]$ and $M_D[i_q]$ apply DCVC-FM-style feature modulation (Li et al., 2024) at the 256-channel interfaces before g_a and after g_s, respectively.

Table 8: Pixel projection and LatentCodec transforms. The partial-convolution residual blocks follow HPCM-Base (Li et al., 2025a), and the quality-indexed feature modulation follows DCVC-FM (Li et al., 2024).

Pixel projection f	**LatentCodec**	
	$M_E[i_q] \to g_a$	$g_s \to M_D[i_q]$
$\mathrm{Conv4{\times}4}(96) \downarrow 2$	$\mathrm{FeatureModulation}(256)$	$\mathrm{TConv2{\times}2}(256) \uparrow 2$
$\mathrm{PConvRB}(96) \times 2$	$\mathrm{PConvRB}(256) \times 4$	$\mathrm{PConvRB}(256) \times 4$
$\mathrm{Conv2{\times}2}(192) \downarrow 2$	$\mathrm{Conv2{\times}2}(320) \downarrow 2$	$\mathrm{FeatureModulation}(256)$
$\mathrm{PConvRB}(192) \times 2$		
$\mathrm{Conv2{\times}2}(256) \downarrow 2$		

Inherited entropy-model components. LatentCodec reuses the AEIC-ME hyper-analysis, hyper-synthesis, and four-pass quadtree context architecture without structural changes (Zhang et al., 2026). This architecture combines a hyperprior (Ballé et al., 2018) with quadtree context coding (Li et al., 2023); its four passes share the context backbone and retain separate input and output adapters. The StarBlock, InceptionNeXt, and gated-CNN implementations follow AEIC-ME and their source architectures (Ma et al., 2024; Yu et al., 2024; Yu & Wang, 2025). The hyperlatent uses an entropy bottleneck, and the spatial latent uses a Gaussian conditional from CompressAI (Bégaint et al., 2020).

Added local-scale modulation. We append a 3×3 convolution with two output channels to the hyper-synthesis output. It produces encoder and decoder scale maps as

$$(\mathbf{s}_E, \mathbf{s}_D) = 0.5 + 1.5\,\sigma(s_{\mathrm{loc}}(h_s))\,.$$

Each map is broadcast over the 320 latent channels. Following the two-map scaling rule of DCVC-FM (Li et al., 2024), $\mathbf{s}_E$ is applied before mean-centered quantization and $\mathbf{s}_D$ after four-pass reconstruction. Both maps are content-conditioned and recovered from the hyperprior without side information, whereas $M_E[i_q]$ and $M_D[i_q]$ are selected by the quality index. After g_s and $M_D[i_q]$, the resulting $\hat{\mathbf{z}}_q$ is the shared compressed representation in Equation 1.

F.2 Decoder Implementation Details

Following AEIC-ME (Zhang et al., 2026), the pruned SD-Turbo U-Net (Sauer et al., 2024) removes time conditioning and text cross-attention. Its trainable interface consists of 3×3 input and output convolutions with channel mappings $256 \to 320$ and $320 \to 256$, respectively, together with rank-32 LoRA adapters. The frozen half-VAE backbone follows AdcSR (Chen et al., 2025) and uses rank-16 LoRA adapters in its attention projections and residual convolutions, with separate input and output convolutions for the two endpoints. All remaining backbone parameters are frozen.

G Sensitivity to the FID Patch Protocol

Figure 1 includes paper-digitized results for Zhou et al. (Zhou et al., 2025) and DIRAC (Ghouse et al., 2023), marked by †. Zhou et al. contribute only rate–PSNR results because FID is not reported for the included configuration. The FID protocol issue therefore concerns DIRAC, whose reported 256×256 patch grid differs from ours. We examine whether this sampling difference changes the perception-endpoint comparison with DIRAC.

Our default protocol uses two grids offset by $(0, 0)$ and $(128, 128)$, yielding 192 patches. We instantiate DIRAC's half-overlapping description with stride 128, yielding 360 patches. We assess this difference by reevaluating the final WIDER checkpoint at all 17 quality indices and both endpoints, keeping the reconstructions, Inception feature extractor, and FID computation unchanged and varying only the patch grid.

Figure 13 shows that absolute FID depends on patch sampling. Relative to the default grid, non-overlapping patches increase P-endpoint FID by 5.58 on average, with a median relative change of 10.54%, whereas 50% overlap decreases it by 6.97 on average, with a median relative change of 12.89%. The mixed-protocol values should therefore remain descriptive rather than be interpreted as a fully harmonized ranking.

Table 9 shows that the P-endpoint result is nevertheless stable under the patch grid described by DIRAC. Matching DIRAC's 50%-overlap grid changes the FID BD-rate from -80.7% to -84.6%.

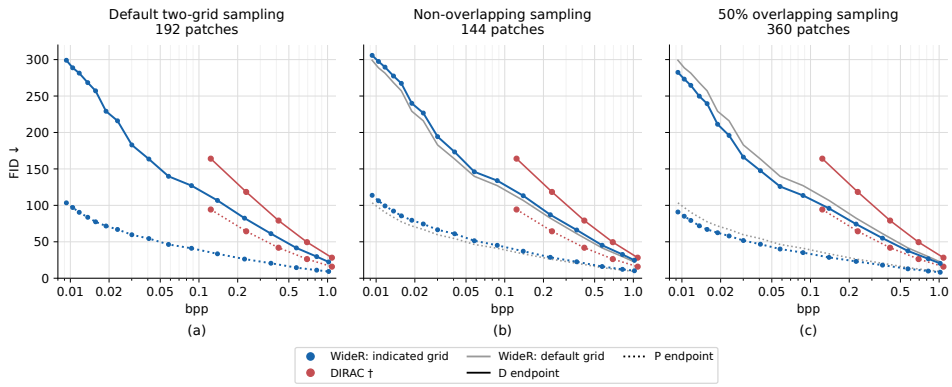


Figure 13: Sensitivity of the DIRAC comparison to FID patch sampling on Kodak24. Panels (a)–(c) evaluate WIDER using the default two-grid protocol, non-overlapping patches, and 50% overlapping patches, respectively. Gray curves in (b) and (c) repeat the default WIDER result to expose the effect of changing only the patch grid. The paper-reported DIRAC curves (†) remain fixed across panels. Solid and dotted curves denote the D and P endpoints. This comparison measures patch-grid sensitivity rather than reproducing the original paper's full FID implementation.

At every paper-reported rate within WIDER's measured range, WIDER has lower P-endpoint FID under the corresponding grid.

Table 9: Kodak24 P-endpoint sensitivity to FID patch sampling. FID BD-rate reports WIDER relative to paper-reported DIRAC results over the same 12 quality indices used in Table 1. Negative values indicate bitrate savings. The last column counts rate-matched comparisons in which WIDER has lower FID; no extrapolation is used.

Baseline	Default grid	Paper-matched grid	Lower FID
DIRAC	-80.7%	-84.6% (50% overlap)	4/4

This paired test isolates patch sampling; it does not establish equivalence between our full FID implementation and that used by DIRAC. The EGIC, CRDR, and CALICE comparisons use their official implementations, whereas MRIC* uses the public Nikolai10/MRIC reimplementation; all four use our common evaluator and are unaffected by this cross-paper protocol analysis.

H FULL ENDPOINT COMPARISONS

Figure 14 expands the CLIC 2020 comparison in Figure 5 to DIV2K and Kodak24, and additionally includes EGIC, CRDR, MRIC*, and CALICE. All curves use the same endpoint-selection rule. All four metrics use a PSNR inset for 0–0.15 bpp, while the perceptual panels directly show the same rate range. FID uses a logarithmic vertical axis. Solid and dotted lines denote single-preference decoding or decoding at the D and P endpoints, respectively. WIDER uses the final checkpoint at 17 quality indices on each dataset. CLIC and DIV2K use the tiled and color-corrected decoding protocol in Section 5.1. Where available, LPIPS (VGG16) and DISTS are evaluated on native-resolution RGB reconstructions and averaged over images.

CALICE (Spadaro et al., 2026) results use the official β-CALICE-STF checkpoint on all three datasets. In CALICE's notation, we evaluate six rate settings $\alpha \in \{0, 0.2, 0.4, 0.6, 0.8, 1\}$ and set $\beta = 0$ and $\beta = 2$ for the distortion and perception endpoints, respectively.

I MORE VISUAL COMPARISONS

We provide additional comparisons with controllable and generative codecs on CLIC 2020 and DIV2K. Every method compresses the complete image, after which identical native-resolution regions are cropped from the decoded images. For the controllable-codec comparisons, EGIC, CRDR, MRIC, and CALICE use their lowest evaluated rate settings; the selected WIDER rate is no greater than every compared per-image rate. The DIV2K examples illustrate texture preservation at lower rates through visual comparison, rather than numerical quality matching.

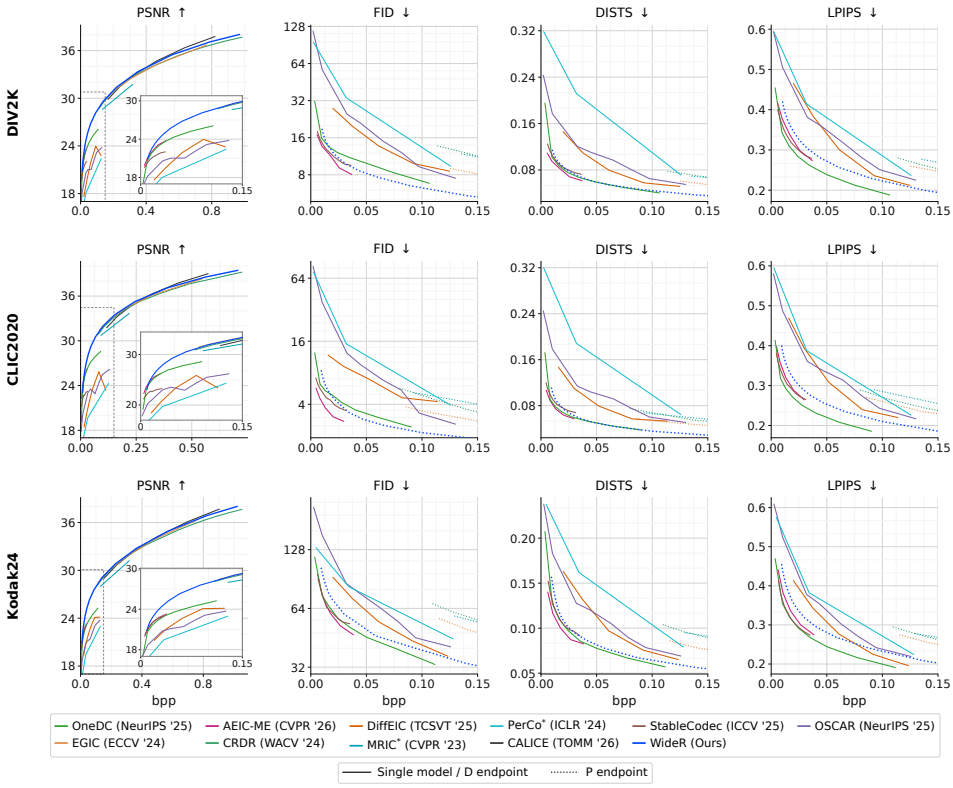


Figure 14: Endpoint comparisons on DIV2K, CLIC 2020, and Kodak24. PSNR uses D endpoints with 0–0.15 bpp insets; perceptual metrics use P endpoints over the same rate range. WIDER uses its final checkpoint; * marks publicly available reimplementations.

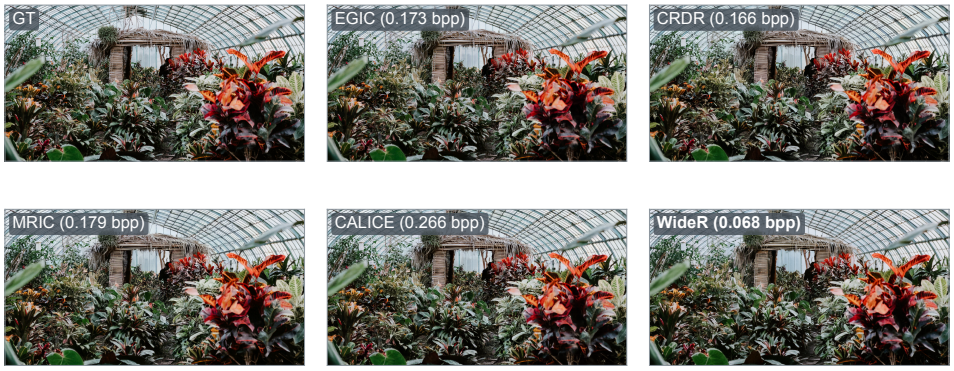


Figure 15: P-endpoint comparison with controllable codecs on CLIC 2020. The wider crop contains dense foliage and repeated greenhouse structure. Values in parentheses are actual full-image rates; WIDER operates at no greater rate than any compared codec.

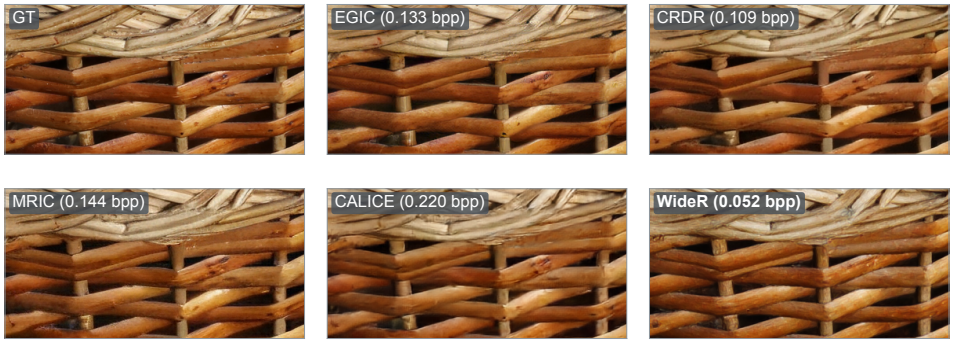


Figure 16: P-endpoint comparison on DIV2K 0822. Identical 384×192 crops show the woven basket after full-image compression. WIDER retains its overall structure and texture with 61% and 53% fewer bits than EGIC and CRDR, respectively. Parentheses report actual full-image rates.



Figure 17: P-endpoint comparison on DIV2K 0880. Identical 384×192 crops show owl feathers after full-image compression. WIDER preserves a comparable appearance of feather texture with 65% and 60% fewer bits than EGIC and CRDR, respectively. Parentheses report actual full-image rates.

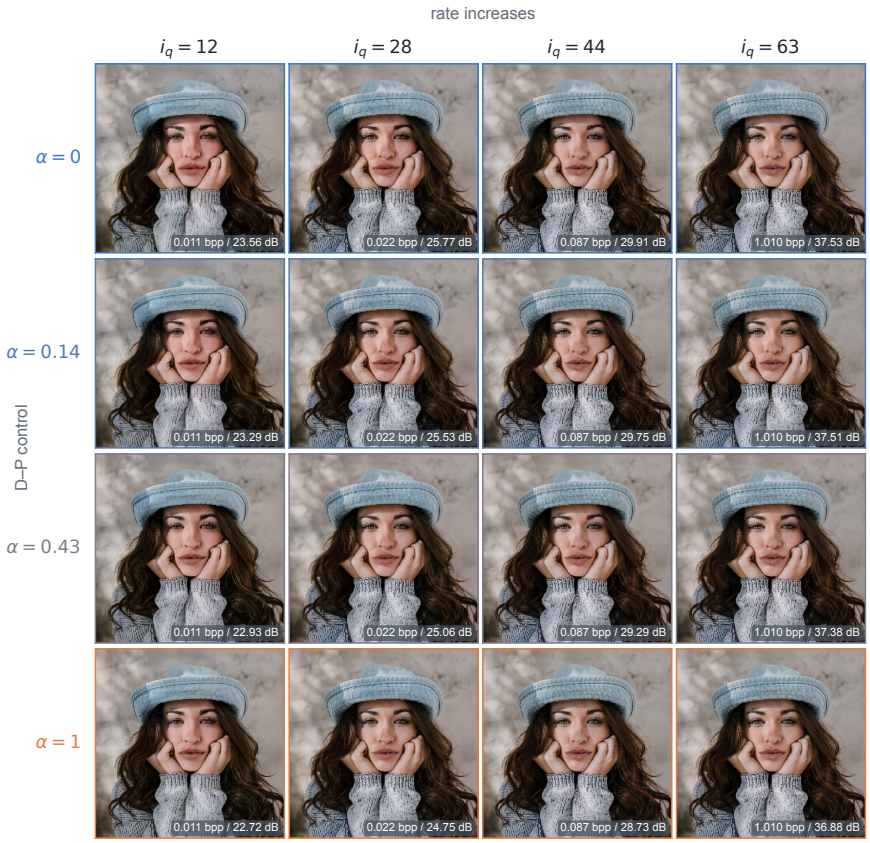


Figure 18: Joint rate and D–P control on DIV2K 0855 with one WIDER model. Columns use quality indices $i_q \in \{12, 28, 44, 63\}$, corresponding to the normalized rate-control parameter $q = i_q/63$, while rows vary the D–P control parameter α from the D endpoint to the P endpoint. Within each column, all reconstructions share the same compressed representation; badges report actual bpp and full-image PSNR.

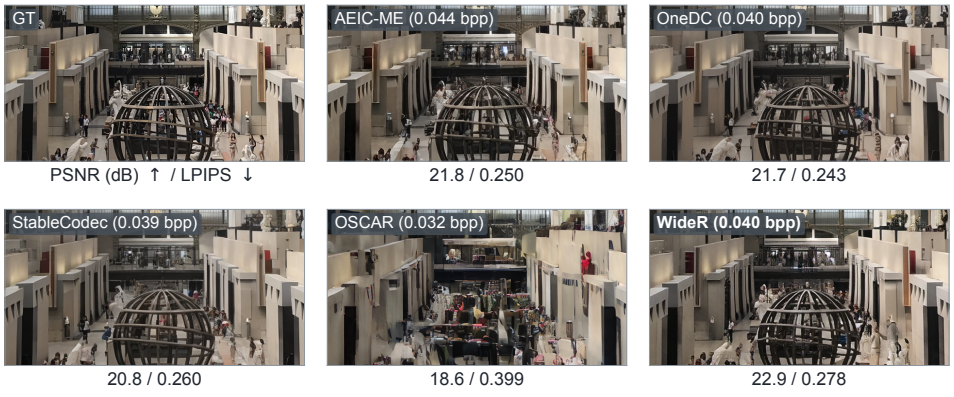


Figure 19: Additional ultra-low-rate comparison with generative codecs on CLIC 2020. The crop contains small people and repeated architectural structure. Rates, PSNR, and LPIPS are measured on the complete decoded image.

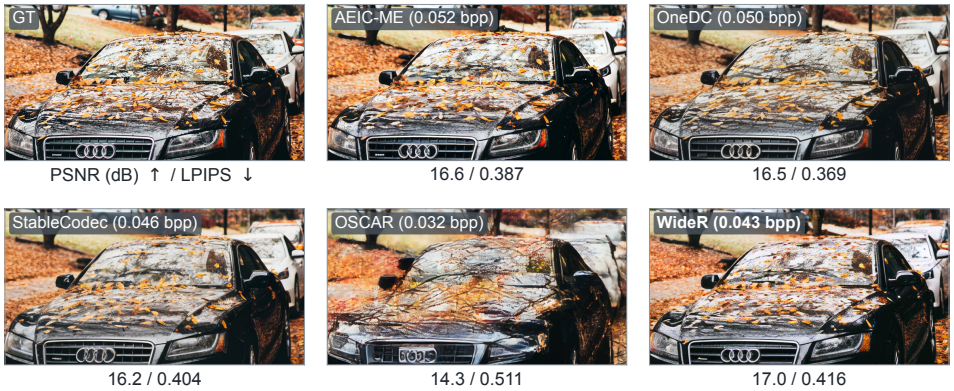
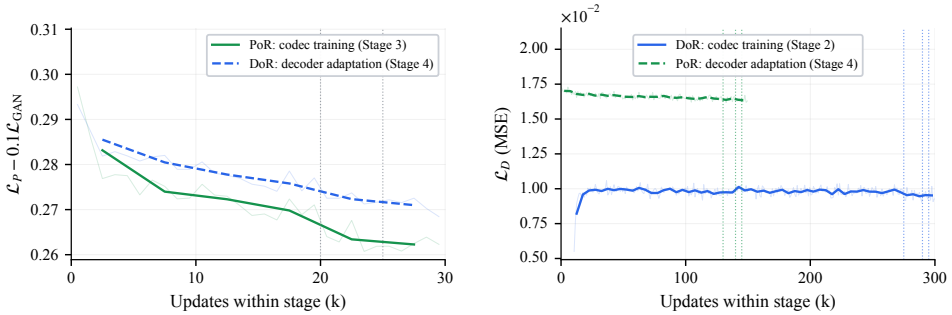


Figure 20: Additional ultra-low-rate comparison with generative codecs on CLIC 2020. The crop combines a license plate, structured vehicle boundaries, and stochastic leaf texture. Rates, PSNR, and LPIPS are measured on the complete decoded image.



(a) P-endpoint refinement

(b) D-endpoint learning

Figure 21: Late-stage training behavior of the DoR and PoR variants. Solid curves show codec training; dashed curves show opposite-endpoint decoder adaptation with the representation frozen. (a) P-endpoint refinement, excluding the GAN term from the plotted loss. (b) D-endpoint MSE, omitting the DoR variant's first 10k gain-only updates. Faint and prominent curves are non-overlapping 1k- and 5k-update means. Axes count updates within each displayed stage, not cumulative training or wall time; vertical dotted lines mark learning-rate reductions.

J Training Budget and Convergence Diagnostics

We examine the completed cross-objective decoder training behind the comparison between the DoR and PoR variants in Section 4.2. This diagnostic does not concern the final WIDER model's intermediate-α training.

Training budget. Following the common 457k-update Stage 1 checkpoint, the DoR and PoR variants each receive 300k codec-training updates. They then freeze f and $\mathrm{LatentCodec}$ and train the opposite-endpoint adaptation modules: the DoR variant uses 120k non-adversarial plus 30k adversarial P-endpoint updates (Stages 3–4), and the PoR variant uses 150k D-endpoint MSE updates (Stage 4). These phases minimize $\mathcal{L}_P$ or $\mathcal{L}_D$ without a rate term or q-dependent multiplier, preserving the representation and original endpoint. Thus, the DoR and PoR variants receive 450k post-warm-up updates, versus 300k for endpoint-only WIDER. These are not equal-compute or equal-sample budgets; Table 5 specifies batch sizes and learning-rate schedules.

Comparable training diagnostics. Figure 21 pairs each decoder-adaptation trace with codec training for the same reconstruction objective. For the P endpoint, we compare Stage 4 of the DoR variant with Stage 3 of the PoR variant using the non-adversarial part of Equation 13:

$$\mathcal{L}_P - 0.1\mathcal{L}_{\mathrm{GAN}} = 2\mathcal{L}_{\mathrm{MSE}} + \mathcal{L}_{\mathrm{DISTS}} + 0.1\mathcal{L}_{\mathrm{CLIP}}.$$

Here $\mathcal{L}_{\mathrm{DISTS}}$ denotes OC-EA-DISTS as in Appendix A.2. The GAN term remains in training; we omit it only from the diagnostic because the discriminators differ. For the D endpoint, we compare Stage 4 of the PoR variant with Stage 2 of the DoR variant using $\mathcal{L}_D = \mathcal{L}_{\mathrm{MSE}}$. All curves exclude rate terms and q-dependent weights. They average sampled training losses, not validation losses; adversarial-phase logs average the final microbatch across ranks, not all accumulation microsteps.

Late-stage behavior. The traces reach their scheduled terminal updates, including all learning-rate reductions. Between each run's final two adjacent 5k-update windows, the P-endpoint diagnostic decreases by 0.490% for the DoR variant's adaptation versus 0.431% for the PoR variant's codec training; D-endpoint MSE decreases by 0.214% for the PoR variant's adaptation versus 0.231% for the DoR variant's codec training, relative to each earlier window mean. Both adaptation traces therefore show small terminal changes, comparable to codec training. Absolute loss levels are not matched-rate comparisons, since codec training changes the representation and rate. These results support completed schedules and late-stage loss stabilization, not global optimality or per-q convergence of FID or PSNR; another adaptation procedure could still improve performance.

K Details of Compared Models

Table 10: GitHub repositories and control parameters of compared models. D and P denote the distortion and perception endpoint settings. * denotes a public reimplementation; † denotes paper-digitized results.

Model	GitHub	R control	D–P control
EGIC (Körber et al., 2024)	Nikolai10/EGIC	–	$\alpha = 0$ (D), 1 (P)
CRDR (Iwai et al., 2024)	iwa-shi/CRDR	$q \in [0, 4]$	$\beta = 0$ (D), 3.84 (P)
CALICE (Spadaro et al., 2026)	EIDOSLAB/CALICE	$\alpha \in [0, 1]$	$\beta = 0$ (D), 2 (P)
MRIC* (Agustsson et al., 2023)	Nikolai10/MRIC	–	$\beta = 0$ (D), 2.56 (P)
PerCo* (Careil et al., 2024)	Nikolai10/PerCo	Target bpp	–
DiffEIC (Li et al., 2025b)	huai-chang/DiffEIC	–	–
StableCodec (Zhang et al., 2025)	LuizScarlet/StableCodec	–	–
OneDC (Xue et al., 2025)	onedc-codec/onedc	–	–
OSCAR (Guo et al., 2025)	jp-guo/OSCAR	Rate level	–
AEIC-ME (Zhang et al., 2026)	LuizScarlet/AEIC	–	–
Zhou et al.† (Zhou et al., 2025)	–	Base-codec rate	$\tau = 1$ (D), 0 (P)
DIRAC† (Ghouse et al., 2023)	–	–	Steps: 1 (D), 100 (P)