ALPHARAE: JOINT COLOR AND OPACITY MODELING WITH REPRESENTATION AUTOENCODERS

Anonymous authors
Paper under double-blind review

ABSTRACT

Transparent images require a representation that preserves foreground color and opacity even when a pretrained visual encoder observes only RGB composites. We introduce AlphaRAE, which augments frozen visual features with a spatial color–opacity code and decodes both through a shared RGBA interface. A conditional adapter supplies this interface from a frozen RGB diffusion transformer. At 1024 pixels, removing spatial detail costs 2.91 dB PSNR, while an alpha-only detail input worsens rendered and direct-opacity fidelity at the reported endpoint. On 434 locally reserved images, the high-resolution model reaches 38.83 dB PSNR and 0.01463 LPIPS, compared with 38.52 dB and 0.02123 for AlphaVAE. The 256-pixel comparison remains mixed, although both configurations reduce direct alpha error on local and external images. At matched latent and trainable budgets, the joint model exceeds an independent-alpha control by 0.60 ± 0.12 dB PSNR across three seeds. Same-RGB generation and recovery studies show that the shared decoder can also receive predicted joint latents; a 53-rater study favors its output over same-alpha color refinement on 64 fixed conditions. These results separate codec fidelity from the ability of a generator to supply its latent.

1 INTRODUCTION

An image asset used in composition must specify more than the colors observed against one background. Its opacity determines whether a hair strand, a reflection, or a partially transparent surface remains plausible when the background changes. This requirement makes RGBA representation learning different from RGB reconstruction: color and opacity must be recoverable together, and their errors interact during rendering. A codec that reconstructs a single composite well can still produce incorrect opacity on another background.

Representation autoencoders (RAEs) pair pretrained visual features with a learned image decoder (Zheng et al., 2026). RAEv2 aggregates encoder layers, while SVG adds residual appearance information for reconstruction (Singh et al., 2026; Shi et al., 2026). AlphaVAE learns a four-channel extension of an RGB VAE (Wang et al., 2025b). Together, these approaches motivate a precise question for a frozen RGB encoder: what transparent information must augment its features, and how can an RGB generator supply that information to the same decoder?

These questions are coupled by the latent interface. An opacity predictor attached to the input image may improve reconstruction while supplying no corresponding variable at generation time. Conversely, directly modifying a pretrained generator to emit a larger latent exposes its color prior to a small RGBA training set. We therefore use an explicit joint code and a mechanism that predicts its complete contents at generation time. Strong reconstruction alone does not establish that the pretrained RGB distribution is compatible with the new decoder.

AlphaRAE makes the missing information explicit in a joint semantic and spatial latent. Opacity is injected into frozen features of the white composite, and a compact spatial encoder receives premultiplied color together with alpha. Its detail code is packed onto the semantic token grid. A shared spatial readout combines the code with decoder context to recover RGB residuals and opacity. Every decoded feature passes through this latent, so the reconstruction and generation interfaces have the same information requirements.

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.

For generation, a conditional adapter predicts the complete joint code from the pretrained RGB generator's output features and pixels. The RGB diffusion transformer and RGBA codec stay fixed. Real RGBA composites and native RGB generations with training-only pseudo opacity provide complementary conditions. At inference, a class label and noise determine an RGB sample, and the adapter supplies the semantic correction and transparent detail required by the RGBA decoder.

We evaluate K7 at 256 pixels and K23 at 1024 pixels against resolution-matched AlphaVAE. On 434 local reserved images, K23 improves PSNR by 0.308 dB and reduces LPIPS by 31%; test SSIM also favors K23, while AlphaVAE retains a higher aesthetic score. At 256 pixels, K7 has a mixed composite ranking. Direct alpha evaluation nevertheless favors both configurations on local and external images. Opaque reconstruction, soft-interior and thin-support subsets, and measured inference costs establish the practical scope of these differences.

The contribution is an explicit RGBA interface and a controlled study of where its transparent information is useful. Learned spatial detail is important; putting appearance into that code improves both rendered fidelity and matte accuracy over an alpha-only alternative. Removing semantic opacity injection reduces rendered fidelity at this endpoint; splitting the small spatial heads has a smaller effect. The K7 downstream study uses a same-alpha refinement control to isolate the foreground-color contribution of joint decoding.

2 Related Work

Representation autoencoders. RAEs pair pretrained representation encoders with learned image decoders, allowing diffusion Transformers (Peebles & Xie, 2023) to operate in semantic feature spaces (Zheng et al., 2026). RAEv2 aggregates multiple encoder layers and introduces changes to representation alignment and guidance (Singh et al., 2026). DINOv3 provides the frozen visual features used here (Siméoni et al., 2025). The distinction between semantic features and reconstruction detail also motivates residual representations in SVG (Shi et al., 2026). AlphaRAE studies this distinction for color and opacity: its added code receives both signals, and its decoder uses the same joint latent for all four output channels. Our K7 and K23 systems inherit different RGB initializers and operate at different resolutions, so their comparison measures complete configurations rather than the isolated effect of layer aggregation.

Transparent image generation. LayerDiffuse adapts latent diffusion to transparency (Zhang & Agrawala, 2024); AlphaVAE extends an RGB VAE into an RGBA codec and evaluates alpha-blended images (Wang et al., 2025b). Qwen-Image-Layered uses an RGBA VAE within a layered-image decomposition system (Yin et al., 2026), while ART combines region layouts with a multi-layer transparent-image autoencoder for text-guided generation (Pu et al., 2025). AlphaRAE studies a single-asset RGBA code built on frozen visual features and a fixed RGB class prior; it neither trains a text generator nor accepts layouts. Its latent budget exceeds AlphaVAE's, which we evaluate on the same images.

Other approaches adapt the generator interface directly. Alfie modifies pretrained DiT inference to obtain RGBA illustrations (Quattrini et al., 2025); TransPixeler uses alpha-specific tokens and LoRA adaptation for video (Wang et al., 2025a). Trans-Adapter targets transparent-image inpainting (Dai et al., 2025). These works address transparent synthesis or editing, while our main quantitative study concerns a representation-autoencoder codec and the interface through which a fixed class generator can supply it.

Matting and spatial reconstruction. Compositing couples foreground color and opacity (Forte & Pitié, 2020); matting spans automatic and guided settings (Li et al., 2023). TransMatting uses trimap-conditioned Transformer tokens (Cai et al., 2023), while Matte Anything uses interactive foundation-model cues (Yao et al., 2024b). Both estimate mattes from observed images, unlike our RGBA codec. ViTMatte reconstructs matting detail (Yao et al., 2024a); learned upsampling addresses boundaries (Dai et al., 2021; Liu et al., 2023). Matting can also learn from coarse trimaps without dense alpha labels (Liu et al., 2025). Our codec uses real RGBA labels; BiRefNet supplies pseudo alpha only during generator-adapter training (Zheng et al., 2024).

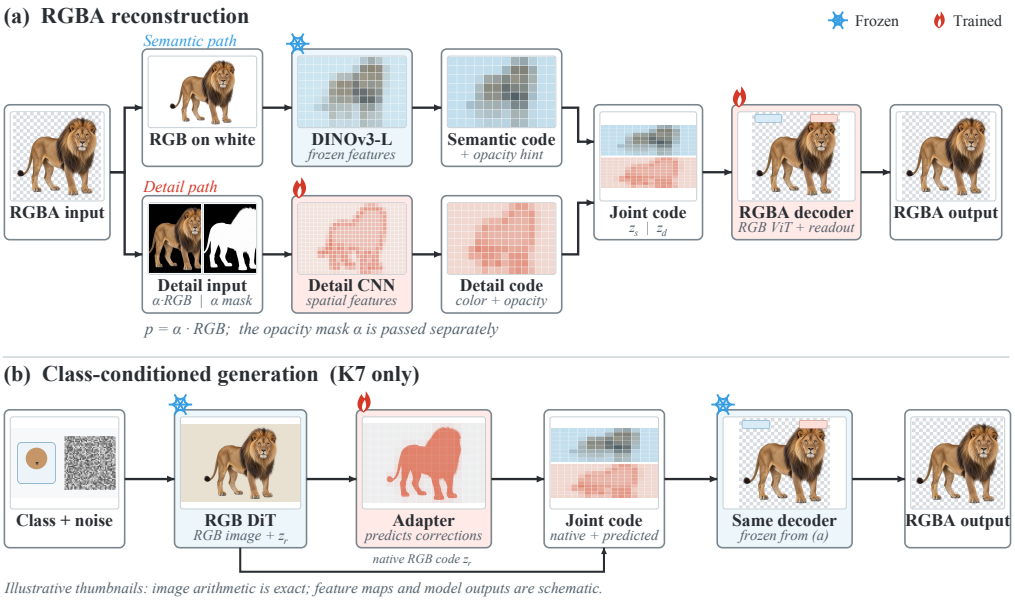

Figure 1: **A shared RGBA interface for reconstruction and generation.** Reconstruction trains the RGB decoder, joint detail encoder, and spatial readout while keeping DINOv3 fixed. Generation freezes the K7 codec and official RGB DiT; a one-step adapter predicts normalized semantic correction and detail codes, which are combined with the native RGB latent for decoding. The teacher used to label synthetic training examples is absent at inference. Thumbnails and feature maps are schematic, not model outputs.

3 AlphaRAE

Following the representation-autoencoder (RAE) construction of Zheng et al. (2026), AlphaRAE retains a frozen pretrained visual encoder and a learned image decoder while extending their RGB interface to one joint RGBA latent. The same codec reconstructs images and receives codes predicted by an adapter from a frozen RGB class generator.

3.1 RGBA Formulation and Joint Representation

Compositing coordinates. Let $x = (f, a)$ denote straight linear RGB $f \in [0,1]^{3\times H\times W}$ and opacity $a \in [0,1]^{1\times H\times W}$. Write $s = S(f)$ for the sRGB transfer of f, and $p = as$ for premultiplied sRGB. The linear-light white composite is $u = af + 1 - a$, with sRGB appearance $w = S(u)$; reconstruction is supervised with sRGB composites, $c_b(x) = as + (1-a)b$.

Shared latent. The white composite cannot identify opacity: a black foreground with $a = 0.5$ and a linear-gray foreground $f = 0.5$ with $a = 1$ produce the same w, but render differently on other backgrounds. AlphaRAE therefore carries premultiplied color and opacity in a decoder-visible detail code alongside semantic tokens. RAEv2 supplies the multi-layer semantic encoder and RGB decoder; the detail stream follows the residual-code pattern of prior autoencoders. Section 4.4 tests semantic opacity injection and spatial-head sharing.

Semantic and spatial encoding. The frozen encoder E_K accepts linear RGB, applies S internally, and then passes the sRGB image through the pretrained visual backbone to produce 1024-dimensional tokens on an $H/16 \times W/16$ grid. Selected RAEv2 layer features are normalized and averaged, then augmented with the spatial mean of the final selected layer. K7 uses zero-based layers $\{11, 13, 15, 17, 19, 21, 23\}$; K23 uses layers 1 through 23. Here K counts aggregated layers, not latent channels.

We inject opacity into the semantic stream and preserve a higher-resolution detail stream. Let U_r be pixel unshuffle by factor r, and let pad_{1024} append zero channels. The encoder is

$$z_s = E_K(u) + \gamma\, \mathrm{pad}_{1024}[U_{16}(1-a)]\,, \qquad \gamma = 0.1, \tag{1}$$

$$z_d = U_2[D_\theta(p,a)]\,, \qquad z = [z_s, z_d]. \tag{2}$$

The spatial encoder D_θ has three stride-two convolutions with channel widths $4, 32, 64, 16$. Its 16-channel output at stride eight becomes 64 channels at stride sixteen. Pixel shuffle exactly reverses this packing during decoding; the learned encoder itself is a lossy representation. The complete latent has 1088 channels, 6.25% more than the RGB RAE's 1024, and contains every feature consumed by the decoder. The detail suffix is part of the decoder input, rather than a mask applied after RGB decoding. Consequently, generation must supply this suffix as well as the semantic features. The RGB generator can retain its native 1024-channel latent as an anchor, while a learned mapping predicts the missing detail and any semantic correction.

Joint decoding. The RGB Transformer decoder produces a base sRGB reconstruction r_0 and token features h. A spatial readout projects h to 64 channels, resizes it to the unpacked detail grid, and fuses it with $U_2^{-1}(z_d)$. Three upsampling blocks produce a four-channel output (u_c, u_a). The reconstructed image is

$$\hat{s} = \mathrm{clip}(r_0 + 0.1u_c, 0, 1), \qquad \hat{a} = \mathrm{sigmoid}(u_a), \qquad \hat{f} = S^{-1}(\hat{s}). \tag{3}$$

Color residuals and opacity therefore share spatial features and token context. The output convolution starts at zero, preserving the RGB initialization and setting initial opacity to 0.5. Reconstruction training updates the entire RGB decoder and the spatial modules; the visual encoder remains frozen. No LoRA modules are used in these reconstruction experiments.

Learned readout interpolation. At stage j, the upsampled feature map is

$$T_j(v) = B(v) + g_j\big(N(v) - B(v)\big), \qquad j \in \{1,2,3\}, \tag{4}$$

where B and N are bilinear and nearest-neighbor $2\times$ upsampling. Each g_j is an unconstrained scalar initialized to zero. The initial operator is bilinear; learned values can also extrapolate beyond a convex mixture. This adds three parameters. Our architectural ablation fixes these scalars to zero while retaining all convolutions and latent dimensions. The gains adapt spatial readout without changing the latent interface or introducing a separate image-space skip path.

3.2 Supervision in Compositing Coordinates

We optimize rendered fidelity together with opacity and local detail. For the nine evaluation colors $\mathcal{B}$, define $\Delta p = \hat{a}\hat{s} - as$ and $\Delta a = \hat{a} - a$. Angle brackets denote a channel-averaged spatial mean over valid pixels. The continuous compositing objective is

$$\mathcal{L}_{\mathrm{comp}} = \frac{1}{9}\sum_{b\in\mathcal{B}} \left\langle (\Delta p - b\Delta a)^2 \right\rangle = \left\langle \Delta p^2 - 2\mu\Delta p\Delta a + \nu\Delta a^2 \right\rangle, \tag{5}$$

where $\mu = (4+q)/9$, $\nu = (4+q^2)/9$, and $q = 128/255$. Opacity broadcasts across RGB channels. The closed form computes the same objective without materializing nine images. Completing the square gives

$$\mathcal{L}_{\mathrm{comp}} = \left\langle (\Delta p - \mu\Delta a)^2 \right\rangle + (\nu - \mu^2)\left\langle \Delta a^2 \right\rangle. \tag{6}$$

Thus compositing already provides an opacity penalty, with coefficient $\nu - \mu^2 \simeq 2/9$. This explains why removing a separate alpha term does not remove alpha supervision. The identity motivates interpretation of the loss ablations; it is an algebraic property of compositing, not an additional learned module.

The remaining terms measure alpha MSE over valid pixels ($\mathcal{L}_\alpha$), alpha MSE on a target-defined edge mask ($\mathcal{L}_{\mathrm{edge}}$), and finite-difference MSE of the residual field $(\Delta p, \Delta a)$ ($\mathcal{L}_{\mathrm{grad}}$). The perceptual term $\mathcal{L}_{\mathrm{perc}}$ averages LPIPS-AlexNet over composites on black and white backgrounds (Zhang et al., 2018). With fixed coefficients, the reconstruction objective is

$$\mathcal{L}_{\mathrm{RAE}} = 1000\mathcal{L}_{\mathrm{comp}} + 200\mathcal{L}_\alpha + 50\mathcal{L}_{\mathrm{edge}} + 80\mathcal{L}_{\mathrm{grad}} + 6\mathcal{L}_{\mathrm{perc}}. \tag{7}$$

Weights do not depend on the observed magnitude of each loss. All training losses operate on continuous images; JPEG is confined to the matched evaluation protocol. At zero opacity, compositing leaves straight RGB unidentifiable, so fidelity is interpreted through opacity and rendered appearance rather than hidden RGB values. Appendix B specifies masks and gradient normalization.

3.3 Class-Conditional Generation Through Latent Adaptation

The additional code in Equation 2 changes the generator interface. We connect it to the official K7 ImageNet DiT by training a separate conditional adapter. Given a class label y and Gaussian noise ϵ, the frozen RGB generator produces an unnormalized native latent $z_r = G(y, \epsilon)$. Its matching frozen RGB decoder produces the sRGB image i_r. For a real RGBA asset $x = (f, a)$, a condition view v composites it in linear light over white or one of two training backgrounds b_v: $i_r^{(v)} = S(af + (1-a)b_v)$ and $z_r^{(v)} = E_K(af + (1-a)b_v)$. Here E_K applies S internally before the frozen backbone, so both expressions describe the same color-space path. Real views require no class label. The adapter A_ψ predicts a semantic correction and the joint detail code. Using the official RGB statistics (μ_r, σ_r) for both condition sources and real-TRAIN detail statistics (μ_d, σ_d), its normalized target is

$$t(x, z_r) = \left[\frac{z_s(x) - z_r}{\sigma_r}, \frac{z_d(x) - \mu_d}{\sigma_d} \right]. \tag{8}$$

Predicted components (δ_s, δ_d) reconstruct the decoder input as

$$\tilde{z} = [z_r + \sigma_r \delta_s,\ \mu_d + \sigma_d \delta_d], \qquad \tilde{x} = \mathrm{Dec}_{\mathrm{RGBA}}(\tilde{z}). \tag{9}$$

For real examples in Equation 8, $z_r = z_r^{(v)}$ and $i_r = i_r^{(v)}$, with one of the three cached views sampled per update. For pseudo-RGBA examples in the final stage, z_r is the frozen DiT's native latent and i_r its RGB decode. The adapter receives normalized RGB tokens and spatially aligned sRGB pixels packed by U_{16}, and predicts both components jointly. The adapter is a one-step supervised predictor: an all-zero 1088-channel state, fixed time $t = 1$, normalized RGB tokens, and $U_{16}(2i_r - 1)$ produce (δ_s, δ_d).

It has eight 384-wide DDT blocks, a two-block 512-wide head, and a 1536-wide spatial residual branch whose output projection starts at zero. The 36.34M-parameter adapter predicts all 1088 channels without direct alpha input; the upstream RGB DiT performs diffusion sampling. Appendix D details the architecture. Training balances semantic and detail MSE and adds decoded-image supervision through the frozen RGBA decoder. Writing t_s, t_d for the target blocks in Equation 8, its supervised objective is

$$\mathcal{L}_{\mathrm{lat}} = \tfrac{1}{2}\,\mathrm{MSE}(\delta_s, t_s) + \tfrac{1}{2}\,\mathrm{MSE}(\delta_d, t_d), \tag{10}$$

$$\mathcal{L}_{\mathrm{adapter}} = \mathcal{L}_{\mathrm{lat}} + \mathcal{L}_{\mathrm{render}}(\tilde{x}, x) + \mathcal{L}_{\mathrm{opaque}}(\tilde{x}, x). \tag{11}$$

Each MSE in Equation 10 averages squared error over the batch, spatial tokens, and channels within its own block (1024 semantic or 64 detail) before the two group means receive equal weight. This prevents the semantic block from numerically overwhelming the detail block. $\mathcal{L}_{\mathrm{render}}$ uses the five terms of Equation 7 through the frozen RGBA decoder, with weights $(10, 2, 0.5, 0.8, 0.06)$ and an all-one valid mask. $\mathcal{L}_{\mathrm{opaque}}$ adds straight-sRGB L1 and 0.25-weighted RGB-gradient L1 where target opacity is at least 0.98. Gradients update only the adapter; neither the RGB DiT nor the RGBA codec is fine-tuned in this stage.

Real RGBA assets supply white and natural-background conditions; native RGB generations receive training-only BiRefNet pseudo alpha. The final pool has 4,499 pseudo images across 32 classes, sampled with 25% probability against 7,779 real assets. Pseudo examples use native DiT latents; real data supervises semitransparent materials. Appendix D gives screening and training history. Inference needs only a class label and seed, not the teacher or a target image.

4 Experiments

4.1 Data, Baselines, and Evaluation

Cohorts. The RGBA corpus contains 8,635 assets supplied with the AlphaVAE dataset. We use a local group split of 7,779 training images, 422 validation images, and 434 reserved test images. Declared groups do not cross splits. This grouping removes exact source duplicates; it does not establish exhaustive visual near-duplicate exclusion. The local partitions differ from AlphaVAE's published benchmark. Exposure in public pretrained models, including AlphaVAE and the RAE backbones, is unverified. Our held-out designation refers to exclusion from AlphaRAE fine-tuning manifests and model selection.

Table 1: **RGBA and opaque RGB test reconstruction.** All methods use the same test cohorts and nine-background JPEG evaluation; opaque RGB has target alpha one. LayerDiffuse and Qwen-Image-Layered use released codecs without diffusion sampling (Zhang & Agrawala, 2024; Yin et al., 2026). Bold marks the best mean per cohort and resolution. AlphaVAE's original training exposure is unverified.

Method	Pixels	PSNR ↑	SSIM ↑	LPIPS ↓	rFID ↓	AES ↑
RGBA local test: 434 images						
LayerDiffuse VAE	256	28.789	0.91148	0.04304	22.425	4.642
AlphaVAE	256	33.710	0.94678	0.02784	23.490	4.661
Qwen-Image-Layered VAE	256	33.880	0.95937	0.02767	19.990	**4.694**
AlphaRAE K7	256	**34.023**	**0.96251**	**0.02679**	**17.724**	4.463
LayerDiffuse VAE	1024	31.800	0.93323	0.04460	10.166	4.983
AlphaVAE	1024	38.523	0.97663	0.02123	3.333	**5.035**
Qwen-Image-Layered VAE	1024	37.916	0.97373	0.01508	2.522	5.013
AlphaRAE K23	1024	**38.831**	**0.98002**	**0.01463**	**2.146**	5.018
Opaque RGB test: 560 images						
LayerDiffuse VAE	256	21.943	0.76769	0.14140	40.073	3.987
AlphaVAE	256	25.628	0.89595	0.09897	46.718	4.041
Qwen-Image-Layered VAE	256	28.333	0.90493	0.07689	29.084	4.018
AlphaRAE K7	256	**30.406**	**0.91915**	**0.05062**	**23.259**	**4.056**
LayerDiffuse VAE	1024	20.470	0.82611	0.11911	21.353	4.332
AlphaVAE	1024	26.116	0.93533	0.07775	20.983	4.339
Qwen-Image-Layered VAE	1024	30.557	0.94740	0.02395	1.415	4.409
AlphaRAE K23	1024	**32.075**	**0.95689**	**0.01633**	**1.292**	**4.411**

Training samples are drawn as 60% original RGBA, 20% opaque composites made with random constant backgrounds, and 20% RGB replay with alpha set to one. The replay pool contains 99,995 images. A separate 560-image opaque RGB validation cohort informed RGB initialization selection. The final RGB test uses a different 560-image source-ID-disjoint split spanning photographs, faces, text, tables, and documents. Both resolutions use the same identities within each test cohort; RGB test images receive fixed center crops with alpha set to one.

Matched reference and capacity. Released AlphaVAE weights use FP32 posterior mode on the same images as our fixed 50,000-update checkpoints. We also compare the released Qwen-Image-Layered RGBA VAE as a codec, without its layer-decomposition generator. K7 starts from official ImageNet RGB weights; K23 inherits 1024-pixel RGB adaptation. AlphaRAE stores $1088(H/16)(W/16)$ latent scalars versus AlphaVAE's $16(H/8)(W/8)$, a $17\times$ difference; active codec parameters are 723.5M versus 83.8M at 1024. Same-A800 FP32 complete-path measurements are 625.3 versus 658.3 ms and 3.49 versus 3.34 GiB. These configuration-level comparisons do not equalize capacity or inherited RGB training (Appendix A).

Metrics. We use AlphaVAE's nine-background JPEG protocol for PSNR, SSIM, LPIPS-AlexNet, reconstruction FID (rFID), and LAION aesthetic score (AES). All codec rows use the same test cohorts, preprocessing, and metric definitions; rFID averages background-specific distances. LPIPS and FID use established features (Zhang et al., 2018; Heusel et al., 2017); AES does not measure opacity. We also measure alpha directly before compositing, separating opacity fidelity from rendered image quality. Adapter validation instead uses continuous images. Bold marks the best displayed mean within a comparable group, not statistical significance.

4.2 Reconstruction of Transparent and Opaque Images

On the RGBA test, K23 reaches 38.831 dB PSNR versus AlphaVAE's 38.523, SSIM 0.98002 versus 0.97663, LPIPS 0.01463 versus 0.02123, and rFID 2.146 versus 3.333 (Table 1). Its 0.308 dB PSNR gain has a paired 95% interval of $[0.164, 0.458]$ dB; AES favors AlphaVAE. K7 improves PSNR, SSIM, and rFID, while its LPIPS interval includes zero and AES favors AlphaVAE. Appendix C gives paired intervals.

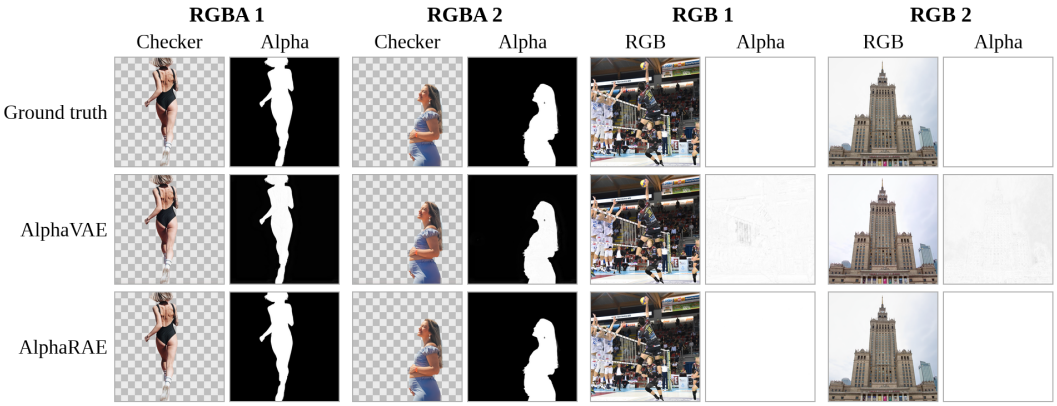

Figure 2: **K23-1024 reconstruction across RGBA and opaque RGB tests.** Rows show ground truth, released AlphaVAE, and AlphaRAE. The RGBA cases (alphavae-00013, alphavae-00059) pair checkerboard composites with alpha; the opaque RGB cases (sa1b_sa_23901, sa1b_sa_23201) pair raw RGB with alpha, whose target is all white. Alpha uses the original fixed grayscale range in every panel. Left to right, per-image nine-background PSNR gains are $+4.29$, $+3.13$, $+0.57$, and $+3.21$ dB. Cases were selected from eight archived previews per cohort where both PSNR and LPIPS favor AlphaRAE; they illustrate differences, not prevalence. No display crop is applied.

On the separate opaque RGB test, K23 reaches 32.075 dB PSNR against AlphaVAE's 26.116 and lowers LPIPS from 0.07775 to 0.01633; all five displayed means favor K23. K7 also improves all five displayed means at 256 pixels. These are fixed-checkpoint composite scores on targets with alpha one, so inaccurate predicted opacity can affect them as well as RGB reconstruction.

LayerDiffuse reaches 31.800 dB PSNR on RGBA 1024 and 20.470 dB on opaque RGB 1024. Qwen-Image-Layered has the highest RGBA-256 AES (4.694); K7/K23 lead PSNR, SSIM, LPIPS, and rFID at 256/1024. These released codecs have different training histories from AlphaRAE; Appendix B gives their evaluation details and Qwen's direct-alpha results.

Figure 2 places K23's transparent and opaque reconstructions next to their targets and AlphaVAE outputs. The RGB color columns show native channels without multiplying by predicted alpha. In both displayed opaque cases, AlphaVAE's matte retains faint scene-shaped gray structure despite an all-white target, whereas AlphaRAE's matte is nearly uniformly white. The raw RGB panels alone would conceal this opacity error.

4.3 Opacity Beyond Composite Fidelity

Table 2: **Direct alpha reconstruction.** FP32 CUDA, fixed 50,000-update checkpoints, identical inputs per resolution. Alpha L1 and its paired 95% difference interval are multiplied by 10^3; lower is better. External AIM excludes exact-file matches against all local splits.

Cohort	Pixels	Images	AlphaVAE	AlphaRAE	Difference interval
Local TEST	256	434	11.399	**4.395**	[-7.763, -6.311]
Local TEST	1024	434	5.261	**1.737**	[-3.884, -3.184]
External AIM	256	499	11.038	**4.220**	[-7.785, -6.032]
External AIM	1024	499	5.030	**1.723**	[-3.731, -2.950]

Composite metrics can favor a method whose opacity is less accurate. Direct continuous-alpha evaluation shows lower Alpha L1 for both configurations than AlphaVAE on the local test and a separately acquired AIM cohort (Table 2). Appendix F also reports alpha MSE, edge L1, pyramid, and gradient errors before compositing. Of 902 released AlphaTest images, 402 exactly match files in the local corpus. Excluding them leaves 499 AIM images and one Internet image; transformed duplicates and pretrained-model exposure remain unverified.

GT-defined soft-interior regions also improve. On 58 external images with substantial soft interiors, K23 reduces region-specific Alpha L1 from 0.04026 to 0.02627; on 68 local test images it decreases from 0.03542 to 0.01775. Counts and membership were fixed from native-resolution GT before inference. Appendix F provides thin-support results, source-category measurements, and interval definitions. These reconstruction measurements do not establish correct transparency in generated images.

4.4 Where Transparent Information Enters

Table 3: **K23 information-path ablations.** All variants use 50,000 updates and the same 422 RGBA validation images (Appendix G). Bold marks the best mean and underlining the second-best. Alpha errors are scaled by 10^3.

Variant	Rendered fidelity		Opacity fidelity ($\times 10^3$)	
	PSNR ↑	LPIPS ↓	Alpha L1 ↓	Edge L1 ↓
Full	**38.488**	**0.01536**	**1.440**	**13.644**
w/o spatial detail	35.577	0.02374	4.402	37.448
w/o opacity injection	37.824	0.02013	1.861	19.082
Separate spatial heads	<u>38.080</u>	<u>0.01782</u>	1.760	17.284
Alpha-only detail	37.955	0.01880	<u>1.562</u>	<u>14.870</u>

Removing spatial detail reduces PSNR by 2.911 dB and worsens alpha error (Table 3). Replacing premultiplied-RGB inputs with repeated alpha at the same nominal width reduces PSNR by 0.533 dB and raises scaled Alpha L1 from 1.440 to 1.562.

Removing semantic opacity injection lowers PSNR by 0.664 dB and raises scaled Alpha L1 from 1.440 to 1.861. Separate spatial heads lower PSNR by 0.408 dB at the displayed endpoint; they still share the joint code and Transformer decoder, with 0.96% fewer readout parameters. Across this endpoint and three additional paired runs, the Full-minus-separate-head PSNR difference is $+0.376 \pm 0.032$ dB (mean and sample standard deviation). Appendix I reports all four pairs and distinguishes this comparison from the independent-alpha control.

Capacity-matched controls. At matched 1088-channel latent and trainable budget, an independent-alpha RGB RAE reaches 38.230 dB PSNR and 2.020 Alpha L1 ($\times 10^3$) on TEST434 versus Full's 38.831 and 1.737. The displayed-checkpoint PSNR gap is 0.601 dB; the three-seed Full-minus-control mean difference is $+0.60 \pm 0.12$ dB. Appendix I specifies training, parameter counts, and the capacity curve.

Supervision and interpolation. The fixed-budget loss study in Appendix C reveals related trade-offs. Removing composition lowers PSNR by 0.662 dB and raises all four direct-alpha errors; fixed bilinear readout lowers PSNR by 0.507 dB and raises Alpha L1 by 37.9% relative to Full. Removing LPIPS worsens RGBA LPIPS by 38.7%. Explicit alpha, edge, and gradient losses improve direct-opacity errors at this endpoint. Their removal slightly improves RGBA LPIPS and AES, but lowers PSNR and SSIM and raises rFID. Equation 6 explains why the composition loss continues to supervise alpha after its explicit loss is removed.

4.5 K7 Class-Conditional Transparent Generation

Figure 3 demonstrates the class-to-RGBA pipeline. The RGB DiT uses 100 Euler steps and internal guidance 1.78; the adapter performs one forward pass. We freeze the terminal checkpoint of the final 8,000-update native-domain stage, with the unchanged 50,000-update K7 codec. The adapter's inherited training path totals 47,500 updates (Appendix D).

The expanded preference study compares 64 same-RGB pairs against the validation-selected BiRefNet foreground refiner (B1) and same-alpha refinement (B2). Across 53 raters in three groups, AlphaRAE's group-mean wins are $(53, 50, 49)$ against B1 and $(47, 49, 48)$ against B2, giving approximately 79% and 75% preference. The two displayed examples illustrate the common RGB



Figure 3: **Frozen same-RGB generation comparison.** Golden retriever and jellyfish show the first seed from one class inside and one class outside pseudo augmentation. B1 refines foreground color from the BiRefNet mask; AlphaRAE predicts the joint latent. Both use the identical frozen RGB DiT image and fixed natural background. The 64-pair preference study compares generated images without ground-truth alpha.

input and a fixed natural background. B1 can preserve sharper RGB texture, while AlphaRAE can change color and opacity. No generated image has a target matte. The B2 comparison favors joint-decoded output under this fixed same-RGB protocol, without measuring generated-alpha accuracy (Appendix J). Difficult classes can still lose thin structures, diffuse opacity, or retain background components.

Table 4: **Frozen RGB-conditioned recovery on external AIM.** The 499 assets yield 998 white/natural observations. B1 refines BiRefNet foreground color; B2 refines color with Ours' alpha. All errors are lower-is-better.

Method	Alpha L1	FG RGB L1	Comp. MSE	Comp. LPIPS
RGB + BiRefNet (B0)	0.02922	0.02712	0.00674	0.06167
BiRefNet + refine (B1)	0.02922	0.02579	0.00646	0.06045
Our alpha + refine (B2)	**0.01554**	0.02418	0.00198	0.05669
AlphaRAE (Ours)	**0.01554**	**0.02308**	**0.00154**	**0.05078**

On external AIM, Ours improves all four metrics over B1 (Table 4). B2 uses the same predicted alpha as Ours, yet Ours lowers foreground RGB L1 from 0.02418 to 0.02308 and composite LPIPS from 0.05669 to 0.05078. This isolates a color-readout contribution to recovery from observed composites, not native generation quality.

5 Discussion and Outlook

The matched-budget independent-alpha control favors Full by 0.60 ± 0.12 dB; increasing detail width from 16 to 64 channels raises TEST434 PSNR from 37.90 to 38.83 dB. AlphaVAE differs in latent and model budgets and inherited RGB training. Direct alpha evaluation tests opacity without color compensation; the same-alpha B2 control fixes opacity to assess foreground color. The adapter study measures same-RGB conversion rather than competition with direct RGBA generators. Text-to-RGBA generation awaits a compatible pretrained text-to-image checkpoint.

6 Conclusion

AlphaRAE joins color and opacity in a decoder-visible RAE latent. Spatial detail and opacity injection improve K23 reconstruction; at matched budget, Full exceeds an independent-alpha control by 0.60 ± 0.12 dB across three seeds. Four paired runs also favor shared over separate spatial readout by 0.376 ± 0.032 dB. Joint-decoded color improves external paired recovery over same-alpha refinement. The 53-rater study yields group-mean wins of $(53, 50, 49)$ against B1 and $(47, 49, 48)$ against B2 across 64 fixed same-RGB pairs. At inference, a one-step adapter supplies the same joint latent to the frozen RGBA decoder. These results establish codec and conditional-conversion gains under distinct protocols.

AI USE STATEMENT

Generative AI tools were used to assist with manuscript drafting. The authors reviewed, revised, and verified all AI-generated text. These tools were not used for experimental design, data collection, or result analysis. All experimental results, claims, and conclusions are based on the authors' own implementations and evaluations. The authors take full responsibility for the final content of this work, including text, claims, and artifacts produced with AI assistance.

REPRODUCIBILITY STATEMENT

Appendices A–D specify the latent interface, color transformations, training schedules, fixed-budget comparisons, data partitions, and metric serialization. AlphaRAE reconstruction rows use 50,000-update checkpoints; AlphaVAE rows use its released weights. The class-generation protocol specifies the frozen models, classes, seeds, and adapter checkpoint. The B1 and B2 preference comparisons each use the same 64 RGB conditions and 53 raters and report approximate mean preferences.

ETHICS STATEMENT

Transparent synthesis can support legitimate editing and compositing as well as misleading visual modifications. Our experiments use existing image corpora and pretrained models, and the generation preference study involves human raters. Individual ratings are kept private; only aggregate preferences are reported. Pseudo opacity labels inherit the foreground teacher's selection biases and are restricted to training. Dataset and model redistribution must follow the respective source licenses and any consent restrictions. The study does not establish provenance or authenticity for generated content.

REFERENCES

Huanqia Cai, Fanglei Xue, Lele Xu, and Lili Guo. Transmatting: Tri-token equipped transformer model for image matting. *arXiv preprint arXiv:2303.06476*, 2023.

Yuekun Dai, Haitian Li, Shangchen Zhou, and Chen Change Loy. Trans-Adapter: A plug-and-play framework for transparent image inpainting. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 15015–15024. IEEE, 2025.

Yutong Dai, Hao Lu, and Chunhua Shen. Learning affinity-aware upsampling for deep image matting. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6841–6850. IEEE, 2021.

Marco Forte. Approximate fast foreground colour estimation. In *2021 IEEE International Conference on Image Processing (ICIP)*, 2021. doi: 10.1109/ICIP42928.2021.9506164.

Marco Forte and François Pitié. F, B, Alpha Matting. *arXiv preprint arXiv:2003.07711*, 2020.

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.

Jizhizi Li, Jing Zhang, and Dacheng Tao. Deep image matting: A comprehensive survey. *arXiv preprint arXiv:2304.04672*, 2023.

Wenze Liu, Hao Lu, Hongtao Fu, and Zhiguo Cao. Learning to upsample by learning to sample. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6027–6037. IEEE, 2023.

Wenze Liu, Zixuan Ye, Hao Lu, Zhiguo Cao, and Xiangyu Yue. Training matting models without alpha labels. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 5604–5612, 2025.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4195–4205. IEEE, 2023.

Yifan Pu, Yiming Zhao, Zhicong Tang, Ruihong Yin, Haoxing Ye, Yuhui Yuan, Dong Chen, Jianmin Bao, Sirui Zhang, Yanbin Wang, et al. Art: Anonymous region transformer for variable multi-layer transparent image generation. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7952–7962. IEEE, 2025.

Fabio Quattrini, Vittorio Pippi, Silvia Cascianelli, and Rita Cucchiara. Alfie: Democratising RGBA image generation with no $$$. In *Computer Vision – ECCV 2024 Workshops*, pp. 38–55. Springer, 2025. doi: 10.1007/978-3-031-92808-6_3.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Minglei Shi, Haolin Wang, Wenzhao Zheng, Ziyang Yuan, Xiaoshi Wu, Xintao Wang, Pengfei Wan, Jie Zhou, and Jiwen Lu. Latent diffusion model without variational autoencoder. In *International Conference on Learning Representations*, 2026.

Oriane Siméoni, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, et al. DINOv3. *arXiv preprint arXiv:2508.10104*, 2025.

Jaskirat Singh, Boyang Zheng, Zongze Wu, Richard Zhang, Eli Shechtman, and Saining Xie. Improved baselines with representation autoencoders. *arXiv preprint arXiv:2605.18324*, 2026.

Luozhou Wang, Yijun Li, Zhifei Chen, Jui-Hsien Wang, Zhifei Zhang, He Zhang, Zhe Lin, and Ying-Cong Chen. TransPixeler: Advancing text-to-video generation with transparency. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18229–18239. IEEE, 2025a.

Zile Wang, Hao Yu, Jiabo Zhan, and Chun Yuan. AlphaVAE: Unified end-to-end RGBA image reconstruction and generation with alpha-aware representation learning. *arXiv preprint arXiv:2507.09308*, 2025b.

Jingfeng Yao, Xinggang Wang, Shusheng Yang, and Baoyuan Wang. ViTMatte: Boosting image matting with pre-trained plain vision transformers. *Information Fusion*, 103:102091, 2024a.

Jingfeng Yao, Xinggang Wang, Lang Ye, and Wenyu Liu. Matte anything: Interactive natural image matting with segment anything model. *Image and Vision Computing*, 147:105067, 2024b.

Shengming Yin, Zekai Zhang, Zecheng Tang, Kaiyuan Gao, Xiao Xu, Kun Yan, Jiahao Li, Yilei Chen, Yuxiang Chen, Heung-Yeung Shum, Lionel M. Ni, Junyang Lin, and Chenfei Wu. Qwen-Image-Layered: Towards inherent editability via layer decomposition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16196–16205, 2026.

Lvmin Zhang and Maneesh Agrawala. Transparent image layer diffusion using latent transparency. *arXiv preprint arXiv:2402.17113*, 2024.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *2018 IEEE/CVF conference on computer vision and pattern recognition*, pp. 586–595. IEEE, 2018.

Boyang Zheng, Nanye Ma, Shengbang Tong, and Saining Xie. Diffusion transformers with representation autoencoders. In *International Conference on Learning Representations*, 2026.

Peng Zheng, Dehong Gao, Deng-Ping Fan, Li Liu, Jorma Laaksonen, Wanli Ouyang, and Nicu Sebe. Bilateral reference for high-resolution dichotomous image segmentation. *arXiv preprint arXiv:2401.03407*, 2024.

A Implementation and Training Details

Representation. Both configurations use DINOv3 ViT-L/16 with 1024-dimensional patch features. Each selected layer is normalized before averaging, and the spatial mean of the final selected layer is added to every token. K7 selects zero-based indices $11, 13, 15, 17, 19, 21, 23$; K23 selects $1, \ldots, 23$. The 64-channel detail code is produced by a three-layer stride-two CNN followed by $2\times$ pixel unshuffle. Decoding applies the exact inverse packing. Packing changes layout, while convolutional encoding changes information content.

The RGB decoder is the supplied 28-layer ViT-XL decoder with hidden width 1152, 16 attention heads, and MLP width 4096. The detail readout projects decoder features to 64 channels, concatenates the unpacked 16-channel code, and applies a 3×3 fusion convolution to 128 channels. The three upsampling blocks have output widths 64, 32, and 16, followed by a four-channel convolution. GELU follows the hidden convolutions. The learned interpolation operators precede those blocks. The output has a scale-0.1 RGB residual and an alpha logit.

Table 5: Representation and trainable parameter budgets. Bold marks fewer latent scalars at the same resolution, not better reconstruction. AlphaVAE uses its released weights; the table does not imply matched training compute. Latent scalar counts exclude batch size.

Configuration	Latent shape	Scalars	Trainable in RGBA SFT
AlphaVAE, 256	$16 \times 32 \times 32$	**16,384**	Reference
AlphaRAE K7, 256	$1088 \times 16 \times 16$	278,528	415,940,071
AlphaVAE, 1024	$16 \times 128 \times 128$	**262,144**	Reference
AlphaRAE K23, 1024	$1088 \times 64 \times 64$	4,456,448	420,363,751

Measured codec cost. All four codecs run on the same A800 in FP32 with TF32 disabled, batch size one, five warmup calls, and three passes over the same first 16 local-test assets. Timed input is a cached CPU RGBA byte array; the complete path includes preprocessing, transfer to the GPU, encoder and decoder inference, transfer back, and output byte conversion. Disk access, PNG encoding, model loading, and metric models are excluded. Peak allocated memory includes codec weights and buffers. Parameter counts include frozen encoders. At 256 pixels the complete paths take similar time, while AlphaRAE uses more memory. At 1024 pixels its measured time is about 5% lower with slightly higher peak memory. These measurements do not isolate capacity or training investment.

Table 6: Complete codec costs under the same batch-one FP32 protocol. Bold marks the lower measured cost within each resolution. Each timing summary contains 48 calls. Parameters include the encoder; memory is peak allocated, not reserved.

Codec	Pixels	Params (M)	Mean ms	p90 ms	GiB
AlphaRAE K7	256	719.09	**60.80**	**64.85**	2.728
AlphaVAE	256	**83.82**	61.20	66.62	**0.510**
AlphaRAE K23	1024	723.52	**625.27**	**628.22**	3.490
AlphaVAE	1024	**83.82**	658.34	660.97	**3.339**

RGB initialization. K7 uses the official matching ImageNet K7 RGB decoder. K23 starts from the general K23 RAEv2 decoder and undergoes direct RGB adaptation at 1024 pixels. Two stages each execute 12,500 updates with global batch 160 and a 99,995-image RGB pool. The first stage uses learning rate 2×10^{-5}; the second uses 10^{-5}. Both optimize RGB L1 and LPIPS-VGG with unit weights and EMA decay 0.99. The second stage inherits the first stage's validation-selected update 11,500; its own terminal EMA at 12,500 initializes RGBA training. Thus the inherited path contains 24,000 RGB updates, while the two executed stages consume 25,000 updates. No RGBA-trained checkpoint or LoRA adapter initializes the reconstruction study.

RGBA optimization. The primary Full run and each single-seed ablation run use 50,000 updates, eight A800 GPUs, one image per GPU, and seed 20260928. Additional stability runs use separate

seeds. AdamW uses $(\beta_1, \beta_2) = (0.9, 0.999)$, weight decay 0.01, and gradient clipping at norm 10. The RGB decoder learning rate is 2×10^{-6}; the joint spatial modules use 2×10^{-4}. After 100 warmup updates, rates remain constant through 35,000, decay by a cosine to 0.1 times their initial values at 45,000, and remain there until 50,000. Training uses BF16 autocast with FP32 loss computation; evaluation is FP32. The reported RGBA checkpoints contain online weights, with no RGBA-stage EMA. The reconstruction studies contain 400,000 nominal image exposures each. Equal updates do not imply equal wall-clock cost between loss ablations.

Data preparation. The RGBA manifest is split into 7,766 training groups, 421 validation groups, and 431 reserved-test groups, containing 7,779, 422, and 434 assets. The split seed is 20260909. Group assignments keep identified exact duplicate source files together. RGB images are converted from source sRGB to straight linear RGB, resized using premultiplied linear color and alpha with antialiased bilinear interpolation, then unpremultiplied where alpha is nonzero. Aspect ratio is preserved and transparent padding completes the square. A valid-pixel mask tracks the nonpadding support. Reconstruction training includes translations of at most eight pixels.

RGB replay comprises 70,000 SA-1B images, 10,000 FFHQ images, 4,995 Chinese-text images, 5,000 English-text images, 5,000 table images, and 5,000 DocLayNet images. The opaque validation cohort contains 160 SA-1B images and 80 from each remaining category. Replay uses random square crops and allowed horizontal flips. Opaque validation uses fixed centered crops, with bicubic resizing when needed. IDs and recorded paths do not overlap between the replay training and validation manifests. The RGB pool audit does not establish comprehensive visual duplicate exclusion.

The 60/20/20 input mixture is sampled during RGBA training. For the opaque-composite branch, each background sRGB channel is independently selected from $\{0, 0.5, 1\}$, and the resulting image's alpha and valid mask are set to one. This augmentation uses 27 possible constant colors; it is distinct from the nine-color loss and evaluation set.

B Loss Definitions and Evaluation Serialization

Let V be the valid-pixel mask and define $\langle e \rangle_V = \sum Ve / \sum V$, with channel averaging when e has several channels. Whole-image alpha loss is

$$\mathcal{L}_\alpha = \left\langle (\hat{a} - a)^2 \right\rangle_V . \tag{12}$$

The edge mask is $M = \mathbf{1}[\mathrm{maxpool}_7(a) - \mathrm{minpool}_7(a) > 0.02]$. The edge loss replaces V by VM. For $e = [\hat{a}\hat{s} - as, \hat{a} - a]$, the gradient objective is

$$\mathcal{L}_{\mathrm{grad}} = \tfrac{1}{2} \left(\langle (\nabla_x e)^2 \rangle_{V_x} + \langle (\nabla_y e)^2 \rangle_{V_y} \right), \tag{13}$$

where the neighbor mask is the product of the valid masks at both pixels. Gradients are adjacent finite differences without a resolution-dependent multiplier. Perceptual loss is

$$\mathcal{L}_{\mathrm{perc}} = \tfrac{1}{2} \sum_{b \in \{0,1\}} \mathrm{LPIPS}_{\mathrm{Alex}}(2c_b(\hat{x}) - 1,\, 2c_b(x) - 1) . \tag{14}$$

LPIPS uses the complete rendered image, including padding. Alpha broadcasts across color channels in the compositing equations.

The nine backgrounds are black, white, gray, red, green, blue, cyan, magenta, and yellow. Gray is byte value 128. For each color channel, four backgrounds contribute value one, four contribute zero, and gray contributes $q = 128/255$. Therefore the first two moments are exactly $(4 + q)/9$ and $(4 + q^2)/9$. Expanding $(\Delta p - b\Delta a)^2$ and averaging yields Equation 5; completing the square yields Equation 6. This derivation holds for the continuous compositing loss. Byte quantization and JPEG do not preserve the identity.

For evaluation, predicted and target RGBA are converted to sRGB and rounded to bytes using clipping of $255x + 0.5$. Byte-normalized RGB and alpha are composited in FP32; the composite is multiplied by 255 and truncated to bytes. PIL's default JPEG encoding uses quality 75. The resulting files are passed to pyiqa 0.1.16 for PSNR, SSIM, LPIPS, FID, and LAION-AES. Reconstruction FID compares predicted and target feature distributions independently on each background; its final score is the arithmetic mean of those nine distances. The metric environment uses Torch 2.10.0, torchvision 0.25.0, NumPy 2.2.6, SciPy 1.16.3, and Pillow 12.1.1.

This serialization is used for the codec comparisons in Table 1, while the differentiable training objective acts before quantization. The protocol reproduces metric operations on the local cohorts, not the original paper's AIM/Alpha subset and subtype aggregation. Reconstruction FID and AES are reported alongside paired metrics to expose different aspects of the tradeoff, rather than combined into a single selection score.

LayerDiffuse reference. The LayerDiffuse VAE reference uses its released SDXL transparent encoder and decoder (Zhang & Agrawala, 2024) with the RealVisXL V4.0 SDXL VAE specified by the official encode–decode demo. The encoder receives the same preprocessed test RGBA quantized to sRGB bytes, applies the official RGB-padding and deterministic transparency offset, and passes its latent to the released decoder. The decoder's eight flip/rotation predictions are combined by the official median rule. Its byte RGBA output enters the same nine-background JPEG evaluator; no diffusion denoising is run. The same released weights are evaluated at 256 and 1024 pixels, and the released lightweight transparent modules differ from the paper's training-time architecture. This is a pretrained configuration reference, not a capacity- or training-matched ablation.

Qwen-Image-Layered reference. The Qwen-Image-Layered row evaluates its released RGBA VAE as an encode–decode codec, without the layer-decomposition generator (Yin et al., 2026). It uses the 434-image RGBA and 560-image opaque RGB test cohorts at both resolutions. The opaque inputs have alpha one; direct Alpha L1 is measured on the reconstructed alpha before compositing. Table 1 reports composite metrics. Direct Alpha L1 ($\times 10^3$) is 5.998 and 4.262 for RGBA 256 and 1024, and 0.409 and 0.559 for opaque RGB 256 and 1024, respectively.

C Complete Component Results

Table 7: Complete K23 loss and interpolation ablations on 422 RGBA validation images. All runs use the same RGB initialization, seed, data mixture, optimizer, and 50,000-update budget. Each loss removal sets one coefficient to zero; fixed bilinear disables the three interpolation gains.

Variant	PSNR ↑	SSIM ↑	LPIPS ↓	rFID ↓	AES ↑
Full	**38.488**	**0.97245**	0.01536	**2.082**	5.042
w/o $\mathcal{L}_{\text{comp}}$	37.826	0.97093	0.01526	2.386	5.042
w/o $\mathcal{L}_{\alpha}$	38.245	0.97156	**0.01514**	2.116	**5.045**
w/o $\mathcal{L}_{\text{edge}}$	38.189	0.97141	0.01531	2.110	**5.045**
w/o $\mathcal{L}_{\text{grad}}$	38.240	0.97148	0.01519	2.141	5.044
w/o $\mathcal{L}_{\text{perc}}$	38.207	0.97193	0.02130	2.356	5.041
Fixed bilinear readout	37.981	0.97000	0.01711	2.412	5.039

Table 8: Exact loss and interpolation settings for the seven K23 runs. Each weight is fixed throughout training. Removing the explicit alpha term retains opacity gradients through other objectives.

Variant	Comp.	Alpha	Edge	Grad.	LPIPS	Interpolation
Full	1000	200	50	80	6	Learned
w/o $\mathcal{L}_{\text{comp}}$	0	200	50	80	6	Learned
w/o $\mathcal{L}_{\alpha}$	1000	0	50	80	6	Learned
w/o $\mathcal{L}_{\text{edge}}$	1000	200	0	80	6	Learned
w/o $\mathcal{L}_{\text{grad}}$	1000	200	50	0	6	Learned
w/o $\mathcal{L}_{\text{perc}}$	1000	200	50	80	0	Learned
Fixed bilinear readout	1000	200	50	80	6	Fixed

The interpolation ablation retains every spatial convolution and fixes the three gains to zero, reducing trainable parameters by exactly three. The five loss removals retain the full architecture and change one coefficient. Configurations share the seed, RGB initialization, training population, batch size, and learning-rate schedule. Removing alpha or gradient supervision does not remove their information from the model input. The alpha-, edge-, and gradient-removal rows have lower RGBA LPIPS and higher AES than Full, but lower PSNR and SSIM and higher rFID. Full also retains better opaque PSNR, SSIM, LPIPS, and AES than the alpha- and gradient-removal arms.

Table 9: All K23 ablations on the 560-image opaque validation cohort, at the same fixed 50,000-update endpoints as Table 7.

Variant	PSNR ↑	SSIM ↑	LPIPS ↓	rFID ↓	AES ↑
Full	**31.943**	**0.95853**	**0.01599**	1.372	4.385
w/o $\mathcal{L}_{\text{comp}}$	31.216	0.95785	0.01618	2.133	**4.392**
w/o $\mathcal{L}_{\alpha}$	31.937	0.95846	0.01600	**1.313**	4.381
w/o $\mathcal{L}_{\text{edge}}$	31.928	0.95845	0.01604	1.333	4.384
w/o $\mathcal{L}_{\text{grad}}$	31.909	0.95820	0.01602	1.352	4.381
w/o $\mathcal{L}_{\text{perc}}$	31.877	0.95791	0.02145	1.459	4.384
Fixed bilinear readout	31.807	0.95789	0.01639	1.513	4.374

Table 10: Direct Alpha errors at the K23 50,000-update endpoints on 422 validation images. Values are multiplied by 10^3 and lower is better. Measurements use decoded continuous FP32 alpha on CUDA, before quantization or compositing.

Variant	Alpha L1	Edge Alpha L1	Pyramid L1	Edge gradient L1
Full	**1.440**	**13.644**	**1.881**	**15.285**
w/o $\mathcal{L}_{\text{comp}}$	1.577	15.261	2.135	16.299
w/o $\mathcal{L}_{\alpha}$	1.624	15.644	2.205	16.527
w/o $\mathcal{L}_{\text{edge}}$	1.710	16.756	2.329	17.281
w/o $\mathcal{L}_{\text{grad}}$	1.556	15.173	2.113	16.271
w/o $\mathcal{L}_{\text{perc}}$	1.587	14.983	2.116	16.089
Fixed bilinear readout	1.986	19.603	2.778	18.854

Alpha L1 is averaged per image over valid nonpadding pixels. Edge Alpha L1 uses the target-derived seven-pixel neighborhood mask from Appendix B; images with no selected edge pixels contribute zero. Edge-gradient L1 averages horizontal and vertical absolute finite-difference errors using adjacent valid edge pixels. Pyramid L1 sums three scales of alpha residual error: each residual subtracts a bilinear-upsampled, factor-two decimation of alpha filtered with the reflect-padded binomial kernel $[1, 4, 6, 4, 1]^{\top}[1, 4, 6, 4, 1]/256$. Later-level valid masks are average-pooled. Each scale is normalized per image before averaging the 422 images. These aggregate diagnostics are recorded in the terminal checkpoint and validation log; per-image diagnostic uncertainty is unavailable.

Table 10 shows that Full has the lowest four direct-alpha means among the displayed arms. Removing the composition term worsens these opacity errors and reduces composite PSNR. Removing gradient supervision also worsens the four direct-alpha means; among RGBA composite metrics, its removal favors only LPIPS and AES over Full. This distinction between rendered fidelity and opacity fidelity motivates reporting both types of evidence.

Table 11: Paired 95% intervals for K23 Full minus AlphaVAE on validation. Positive is better for PSNR, SSIM, and AES; negative is better for LPIPS. Resampling keeps each asset's nine backgrounds together.

Cohort	Metric	Mean difference	95% interval
rgba422	PSNR	+0.17957	[+0.03088, +0.34212]
rgba422	SSIM	-0.00262	[-0.00320, -0.00199]
rgba422	LPIPS	-0.00727	[-0.00824, -0.00633]
rgba422	AES	-0.01509	[-0.02209, -0.00776]
opaque560	PSNR	+6.55959	[+6.23500, +6.87626]
opaque560	SSIM	+0.02908	[+0.02692, +0.03137]
opaque560	LPIPS	-0.07148	[-0.07516, -0.06794]
opaque560	AES	+0.08706	[+0.07062, +0.10386]

Intervals use 5,000 bootstrap draws with seed 20260922: 421 source groups for RGBA and 560 assets for opaque RGB. Group resampling preserves asset-weighted means when group sizes differ. These are pointwise intervals for fixed checkpoints, not simultaneous intervals over the ablation family or repeated-training-seed uncertainty. No rFID bootstrap interval is estimated. On RGBA

validation, 40.76% of individual assets have higher PSNR with K23 Full, while 90.05% have lower LPIPS. Thus the positive mean PSNR difference is not a per-image dominance claim.

Table 12: Reserved-partition differences, AlphaRAE minus resolution-matched AlphaVAE. The displayed 95% intervals accompany the fixed-checkpoint means; resampling respects declared source groups.

Configuration	Metric	Mean difference	95% interval
K7	PSNR	+0.31300	[+0.01383, +0.62423]
K7	LPIPS	-0.00106	[-0.00315, +0.00136]
K7	AES	-0.19844	[-0.21401, -0.18257]
K23	PSNR	+0.30797	[+0.16374, +0.45801]
K23	SSIM	+0.00339	[+0.00180, +0.00504]
K23	LPIPS	-0.00660	[-0.00747, -0.00576]
K23	AES	-0.01687	[-0.02382, -0.01003]

K23 improves per-image PSNR on 44.24% of reserved assets and LPIPS on 90.09%; its positive mean PSNR does not imply per-image dominance. K7's LPIPS difference is -0.00106 with interval $[-0.00315, 0.00136]$.

D Conditional Adapter Training and Sampling

The adapter uses spatial concatenation in both DDT input projections. Its inputs comprise 1088 zero-state residual channels, 1024 normalized RGB feature channels, and 768 channels obtained by $16\times$ pixel unshuffle of RGB pixels scaled to $[-1, 1]$. Output width is 1088. Attention head counts are six and eight for the 384- and 512-wide block groups. The class-conditioning token is fixed; ImageNet class information reaches the adapter through the frozen RGB generator's features and image, not a separately supplied class token.

The final model adds a spatial correction branch to the DDT prediction. It concatenates the 1088 predicted residual channels with 1024 normalized RGB feature and 768 packed-pixel channels. A 1×1 convolution projects to width 1536, followed by 32-group normalization and SiLU. A depthwise 3×3 convolution and pointwise 1×1 mixing form a residual block. The final 1×1 projection produces a 1088-channel correction added to the original prediction. This projection starts at zero when the branch is introduced. Total adapter parameters increase from 27,860,116 to 36,336,852; reconstruction-codec weights are unchanged.

Real-condition caching uses each training RGBA asset's white composite and two composites over training SA-1B backgrounds. Targets come from the same frozen K7 RGBA encoder used to define the generation interface. Per-channel detail means and standard deviations are fitted on real training assets, with standard deviations floored at 0.01. RGB normalization uses the supplied mean and $\sqrt{\text{variance} + 10^{-5}}$. Training adds Gaussian noise with standard deviation 0.02 to normalized RGB feature conditions. Semantic and detail target MSEs are averaged within their own channel groups and weighted equally.

Decoded-image losses backpropagate through the frozen RGBA decoder into the adapter. Their weights are composition 10, alpha 2, edge 0.5, gradient 0.8, and LPIPS 0.06. The valid mask is all ones, including padding. Early stages use four latent examples but one decoded-image example per rank; the color-fidelity stage and all later stages decode all four. Those stages also add straight-sRGB L1 with weight one inside target $a \geq 0.98$, and horizontal/vertical RGB-gradient L1 with weight 0.25 where both neighboring target pixels meet this threshold. Mask normalization is per image; predicted alpha does not gate these color losses. The head uses BF16 autocast, with FP32 decoding and losses. AdamW uses betas $(0.9, 0.95)$, weight decay 0.01, clipping at norm one, and EMA decay 0.99.

The final adapter has 47,500 updates along its inherited path. Adding pixel input channels zero-initializes their projection weights, preserving the preceding mapping. Each stage uses 100 warmup updates and cosine decay over its final 20% to one tenth of the peak rate. A real-only continuation and a pseudo-mixture arm each run 1,500 updates from the same image-only checkpoint. Their conditional validation PSNR is 26.727 and 26.689 dB, with LPIPS 0.09790 and 0.09805. The mixture

Table 13: Inherited adapter training path. Each stage starts from the preceding selected EMA with a new optimizer. Sequential stages are not equal-compute ablations. Paired learning rates denote inherited head and added refiner.

Stage	Updates	Batch	Peak LR	Change
Spatial conditional SFT	3,000	96	10^{-4}	RGB feature condition
Pixel-conditioned SFT	1,500	96	5×10^{-5}	Add packed RGB pixels
Image-only continuation	1,500	96	3×10^{-5}	Same architecture
Native-domain mixture	1,500	96	3×10^{-5}	Add pseudo RGBA pairs
Long continuation	20,000	192	10^{-5}	Same mixed data and objective
Color fidelity	4,000	192	10^{-5}	Full decoded batch, RGB losses
Wide residual refinement	8,000	192	$3\times10^{-6}/10^{-5}$	Add 1536-wide branch
Native-domain expansion	8,000	192	$3\times10^{-6}/10^{-5}$	Larger pseudo pool, native conditions

was selected using inspected native-generation behavior. These paired-validation measurements are not class-generation scores.

The 20,000-update continuation reaches 27.563 dB conditional PSNR, 0.01423 Alpha L1, and 0.07775 LPIPS on 844 validation pairs. The final native-domain endpoint reaches 28.285 dB, 0.01429, and 0.06693. Color supervision, the residual branch, more decoded examples, and later data expansion enter sequentially, so this change does not isolate a single intervention. The displayed model is the last completed endpoint, frozen before its current baseline evaluation. Earlier results on the 434 local images informed later color-fidelity development; their new adapter evaluation is therefore development-reused, even though the images never entered fine-tuning.

Table 14: Matched alternatives within adapter development stages. Scores use 422 real validation assets with two RGB conditions per asset, continuous sRGB composites, and no JPEG. Each pair shares its initialization and update budget; different pairs inherit different checkpoints.

Pair	Alternative	Updates	PSNR ↑	Alpha L1 ↓	LPIPS ↓
Initial	Single step	3,000	**24.297**	**0.02017**	**0.13592**
Initial	Conditional Flow	3,000	22.201	0.02304	0.14918
Color loss	Ordinary	1,500	25.983	0.01700	**0.11326**
Color loss	GT-alpha factor	1,500	**26.039**	**0.01689**	0.11795
Text	Image only	1,500	**26.455**	**0.01573**	**0.10365**
Text	Image + caption	1,500	26.454	**0.01573**	**0.10365**
Data	Real only	1,500	**26.727**	**0.01518**	**0.09790**
Data	Native mixture	1,500	26.689	0.01552	0.09805

The initial comparison shares a 250-update supervised bootstrap before the Flow arm switches to conditional flow matching and 50-step sampling. The color-loss pair starts from the 3,000-update single-step EMA; both arms add pixel conditions. GT-alpha factorization changes color supervision during training, with the same ordinary evaluation loss and no ground-truth alpha input at inference. The text pair starts from the ordinary pixel-conditioned EMA and compares image conditions with additional frozen CLIP text features (Radford et al., 2021); real validation uses image descriptions for that arm. The data pair starts from the image-only continuation EMA. These alternatives motivate the chosen path, but comparisons between different pairs conflate continued training with changed settings.

The native-domain set uses 64 seeds for each of eight classes: brown bear, cheetah, leopard, lion, zebra, daisy, teapot, and sports car. A frozen BiRefNet model supplies pseudo alpha. Six of 512 examples are rejected when the fraction of pixels above alpha 0.5 falls outside $(0.01, 0.98)$. The remaining 506 examples are each sampled with multiplicity five, alongside 7,779 real examples. Their three condition views are the native RGB latent, the re-encoded native RGB image, and the white composite of the pseudo RGBA. The pseudo masks can retain multiple foreground objects and do not identify class-exclusive or physically correct alpha.

The final expansion generates 128 additional seeds for each of 32 classes, disjoint from monitoring and earlier pseudo seeds. It retains 3,993 of 4,096 candidates using the same area criterion and

requiring that at most 15% of pixels have rounded alpha in $(0.1, 0.9)$. Together with the prior 506 images, the final pool has 4,499 examples. Deterministic weighted draws with replacement assign 75% probability to real data and 25% to pseudo data. Pseudo examples then use their native RGB DiT latent exclusively; real examples retain all three observation choices. This changes condition sampling as well as the unique data count. The 32-class pool covers the eight original classes plus additional mammals, household objects, instruments, fruit, vegetables, and vehicles. It continues to exclude the bubble, jellyfish, spider-web, and plastic-bag classes in the difficult gallery.

Figure 3 shows golden retriever (class 207, inside pseudo augmentation) and jellyfish (class 107, outside it) with seed 20261001. The preference study uses 64 same-RGB pairs evaluated by 53 raters in three groups. AlphaRAE receives approximately 79% preference against B1 and 75% against same-alpha B2. The outside-pseudo example is not semantic OOD. The RGB generator uses the official K7 EMA, 100 Euler steps, x prediction, logit-normal time shift eight, $t_\epsilon = 0.05$, CFG one, and internal guidance 1.78 active over $[0.1, 1]$. The final pipeline unnormalizes the native RGB latent, decodes its condition, runs the adapter once, and invokes the frozen RGBA decoder. Generated images have no ground-truth alpha.

E Qualitative Reconstruction and Generation

The RGBA reconstruction plates use the first two manifest-ordered reserved-test previews, without sorting by metric or visual quality. They display both alpha and compositing behavior. All model PNGs come from the matched FP32 CUDA evaluations used for Table 1. No alpha contrast enhancement or spatial crop is applied to these plates.

F Direct Opacity And External Evaluation

All four models are evaluated sequentially on the same CUDA device in FP32, without autocast. Inputs use the established aspect-fit linear-premultiplied preprocessing. Predicted alpha is evaluated before quantization, background composition, or JPEG. K7 and K23 use their fixed 50,000-update reconstruction checkpoints; AlphaVAE uses its released weights and posterior mode. Each local-test image is held out from our fine-tuning manifests. The external release contains 500 AIM-derived and 402 Internet images before screening. Exact byte comparison with all local TRAIN, VAL, and TEST sources excludes one AIM image and 401 Internet images. The retained set has 499 AIM images and one Internet image. This establishes exact-file exclusion, not exclusion of transformed copies or unrecorded source relationships.

Subset membership is computed from canonical 1024-resolution target alpha before model evaluation and shared across both tested resolutions. Support is $a > 0.01$. Eroding this support with a 17×17 window defines its interior. A soft-interior image has at least 1% of valid pixels and 10% of support pixels with $0.05 < a < 0.95$ inside the eroded support. A thin-support image has at least 1% foreground support and loses at least 10% of that support under a 17×17 morphological opening. The local set contains 68 soft-interior and 14 thin-support images, with eight in both. The external set contains 58 and 34, with 18 in both. These are geometry proxies; they are not manually validated glass, liquid, or hair categories.

The external source's *transparent* label contains 33 retained examples. It includes overlays and does not exclusively identify physical transparent materials. Thin-support proxy membership excludes all portraits at this threshold, so the corresponding results cannot represent all hair cases. We retain the source-category and geometry-proxy analyses separately.

Whole-image errors on a difficult subset still include easy background pixels. We therefore also evaluate only the soft-interior or thin-support pixels within each selected image. Regional masks are recomputed at the evaluated resolution with radius eight at 1024 and radius two at 256. Empty regions are omitted, with counts retained. In contrast, the established edge metric retains a seven-pixel window at each resolution. Cross-resolution error differences therefore cannot isolate the encoder aggregation effect.

Pointwise intervals use 5,000 paired bootstrap draws and seed 20260923. Local samples are resampled by the 431 declared groups. External resampling groups exact images; original AIM scene groups are unavailable, so these intervals do not account for unrecorded scene dependence. They



Figure 4: K7-256 RGBA reconstruction: ground truth, released AlphaVAE, and AlphaRAE on the first two reserved-test assets. Alpha panels expose opacity structure separately from checkerboard composites.



Figure 5: K23-1024 reconstruction on the same two reserved-test assets. Both models' PNGs come from the matched FP32 CUDA evaluation. AlphaRAE uses the fixed full-method checkpoint at 50,000 updates.

Table 15: Direct alpha errors on predefined cohorts and subsets, multiplied by 10^3. Every metric is lower-is-better. Mask definitions follow Appendix B; pyramid and edge-gradient definitions are in Appendix C.

Cohort	Pixels	n	Model	L1	MSE	Edge L1	Pyramid	Gradient
Local TEST	256	434	AlphaVAE	11.399	1.706	47.574	17.152	39.250
Local TEST	256	434	AlphaRAE	**4.395**	**0.466**	**18.537**	**6.332**	**20.584**
Local TEST	1024	434	AlphaVAE	5.261	0.345	30.542	6.081	23.561
Local TEST	1024	434	AlphaRAE	**1.737**	**0.104**	**14.644**	**2.372**	**15.571**
Local soft	256	68	AlphaVAE	25.633	3.739	59.738	33.043	37.630
Local soft	256	68	AlphaRAE	**11.437**	**1.203**	**26.788**	**15.787**	**26.439**
Local soft	1024	68	AlphaVAE	11.611	0.836	33.301	12.852	19.628
Local soft	1024	68	AlphaRAE	**4.715**	**0.237**	**15.499**	**6.333**	**14.691**
Local thin	256	14	AlphaVAE	55.895	12.711	86.757	67.903	72.799
Local thin	256	14	AlphaRAE	**35.588**	**6.376**	**52.942**	**46.753**	**58.810**
Local thin	1024	14	AlphaVAE	18.631	2.165	42.075	26.759	35.391
Local thin	1024	14	AlphaRAE	**11.858**	**1.200**	**25.803**	**16.950**	**26.716**
External AIM	256	499	AlphaVAE	11.038	1.574	44.848	16.505	37.521
External AIM	256	499	AlphaRAE	**4.220**	**0.378**	**19.029**	**6.207**	**21.477**
External AIM	1024	499	AlphaVAE	5.030	0.339	29.781	6.023	22.334
External AIM	1024	499	AlphaRAE	**1.723**	**0.093**	**14.435**	**2.380**	**15.090**
External soft	256	58	AlphaVAE	28.937	3.859	61.722	34.357	41.721
External soft	256	58	AlphaRAE	**14.457**	**1.385**	**33.036**	**19.560**	**35.430**
External soft	1024	58	AlphaVAE	13.217	0.972	32.477	15.224	19.569
External soft	1024	58	AlphaRAE	**6.930**	**0.429**	**19.715**	**9.567**	**17.870**
External thin	256	34	AlphaVAE	41.112	7.715	84.571	48.859	57.331
External thin	256	34	AlphaRAE	**15.680**	**1.816**	**35.394**	**21.851**	**39.258**
External thin	1024	34	AlphaVAE	16.781	1.837	42.698	20.646	27.625
External thin	1024	34	AlphaRAE	**6.765**	**0.475**	**20.444**	**9.489**	**19.765**
Transparent label	256	33	AlphaVAE	36.940	5.785	65.711	40.156	39.480
Transparent label	256	33	AlphaRAE	**14.082**	**1.225**	**28.823**	**19.478**	**30.438**
Transparent label	1024	33	AlphaVAE	15.168	1.343	32.759	16.921	18.135
Transparent label	1024	33	AlphaRAE	**5.909**	**0.273**	**16.165**	**8.202**	**14.424**

Table 16: Region-restricted alpha L1 on GT-defined subset images. Difference intervals are AlphaRAE minus AlphaVAE; no image/background pixels outside the named region contribute.

Region subset	Pixels	n	AlphaVAE	AlphaRAE	95% interval
Local Soft interior	256	68	0.07334	**0.03932**	[-0.04011, -0.02821]
Local Soft interior	1024	68	0.03542	**0.01775**	[-0.02181, -0.01384]
Local Thin support	256	13	0.10792	**0.05584**	[-0.07160, -0.03383]
Local Thin support	1024	14	0.06206	**0.03661**	[-0.03328, -0.01783]
External Soft interior	256	58	0.07856	**0.05305**	[-0.02991, -0.02137]
External Soft interior	1024	58	0.04026	**0.02627**	[-0.01695, -0.01136]
External Thin support	256	34	0.11037	**0.03882**	[-0.08725, -0.05730]
External Thin support	1024	34	0.04856	**0.01876**	[-0.03659, -0.02364]

condition on the evaluated checkpoints rather than repeated training seeds. The source *transparent* subset and the small thin-support subsets should be interpreted with their sample counts, not as evidence of universal material handling.

G Information-Path Controls

The five structural variants use the K23 RGB initializer, the same 7,779 training and 422 validation asset identities, a 60/20/20 input mixture, five fixed loss weights, global batch eight, and a 50,000-update schedule. Table 3 compares Full with the no-detail, no-injection, separate-head, and alpha-only controls. PSNR/LPIPS refer to JPEG composites; alpha errors use continuous predictions as in Appendix C.

The no-detail model removes its learned encoder and supplies 64 zero channels to the retained spatial readout, preserving the tensor interface but reducing informative capacity. The no-injection model sets the effective γ in Equation 1 to zero and retains the learned detail stream. Alpha-only detail repeats alpha into the four input channels of the same CNN. It matches the nominal parameter and latent budgets while changing available information and the effective first-layer parameterization.

Separate readout uses two independent small spatial branches for RGB and alpha, with widths 95, 47, 23, and 11 and 32 context channels each. Their combined readout has 261,012 parameters versus Full's 263,543, and six learned interpolation gains versus three. The joint code and upstream RGB Transformer decoder remain shared. This control tests small-readout sharing, not two independent full decoders.

Full, no-detail, no-injection, and alpha-only detail have 420,363,751, 420,334,839, 420,363,751, and 420,363,751 trainable parameters, respectively; separate spatial heads have 420,361,220. Every variant retains a $1088 \times 64 \times 64$ external latent shape, although the no-detail suffix is zero and carries no learned information. Table 3 compares the common PSNR, LPIPS, whole-image alpha, and edge-alpha metrics without mixing auxiliary diagnostics across distinct runs.

The four-metric comparison shows lower rendered and opacity fidelity without semantic opacity injection or with alpha-only detail. A separate Full run with seed 20260929 records 38.376 dB PSNR versus 38.488 dB for the displayed Full seed. This 0.112 dB Full-only spread is descriptive: seed 20260929 has no matched separate-head run and is excluded from the four paired runs below.

H Same-Source Foreground Baseline

The comparison fixes the pretrained RGB DiT and its output image. The original B0 baseline predicts opacity with the official BiRefNet checkpoint and retains the observed RGB values. The follow-up B1 control refines foreground color from that same mask; B2 applies the same refiner to AlphaRAE's predicted alpha as a same-alpha diagnostic. AlphaRAE uses the terminal 8,000-update native-domain adapter stage, whose inherited path totals 47,500 updates, with the frozen K7 RGBA decoder. BiRefNet also supplied the 4,499 pseudo-alpha training examples; its labels and biases are part of the adapter's training history.

All 48 monitoring cases and eight fixed examples are retained, with identical class IDs, seeds, and RGB pixels. Forty cases belong to classes present in the final 32-class pseudo pool; 16 belong to the four other sampled classes. The RGB generator already supports those ImageNet classes, and real RGBA training is not class-disjoint. The baseline has cleaner cutouts or sharper texture in several examples. In transparent-material cases the teacher often produces nearly opaque masks, whereas the adapter can produce diffuse or context-dependent alpha. Neither behavior establishes physical transparency without a target matte.

Paired quantitative task. For each of the 434 local assets, we construct two RGB observations by linear-light composition followed by sRGB conversion and byte rounding, using white and one deterministic SA-1B training background. Both methods receive identical observations; GT alpha is used to construct and evaluate the pair, not as model input. The adapter encodes observed RGB with frozen DINO, whereas native generation already provides its RGB latent. These images remain excluded from fine-tuning, but their earlier aggregate baseline results informed subsequent color/refiner development. The current local comparison is therefore development-reused.

Alpha and premultiplied-color errors use the valid image region. Edge Alpha L1 uses the seven-pixel target-alpha neighborhood mask. Foreground RGB L1 weights straight sRGB error by target alpha. The baseline retains the observed composite RGB and does not perform foreground-color unmixing. Nine-background MSE uses continuous sRGB composites without JPEG. On the local pairs, both premultiplied-color and foreground-RGB differences favor AlphaRAE.

External paired evaluation and controls. The final frozen head is evaluated on the retained external cohort of 499 AIM assets and one Internet asset. Table 4 reports AIM separately: each source supplies a white and a deterministic natural-background observation, yielding 998 paired RGB conditions. The same observed RGB is passed to every method; target RGBA is used to construct and score those observations, not as model input. Public-model pretraining overlap is unverified.

Table 17: Complete local paired results on 868 development-reused observations. Differences are AlphaRAE minus BiRefNet; negative is better. Pointwise 95% intervals use 2,000 bootstrap draws over 431 declared groups, retaining both views.

Error	BiRefNet	AlphaRAE	Difference	95% interval
Alpha L1	0.041811	**0.018225**	-0.023586	[-0.029521, -0.017985]
Alpha MSE	0.030038	**0.007191**	-0.022847	[-0.028686, -0.017399]
Edge Alpha L1	0.099613	**0.072467**	-0.027146	[-0.034728, -0.020296]
Premultiplied RGB L1	0.025416	**0.020226**	-0.005190	[-0.008870, -0.001606]
Foreground RGB L1	0.039074	**0.022800**	-0.016274	[-0.018862, -0.013688]
Composite MSE	0.010419	**0.003142**	-0.007277	[-0.009108, -0.005513]

The BiRefNet foreground refiner's radius was selected from $\{30, 60, 90, 120\}$ by foreground RGB L1 on 422 local validation assets; radius 30 was then fixed. B1 and B2 use the same `refine_foreground` routine from the BiRefNet deployment handler (`https://huggingface.co/not-lain/BiRefNet/blob/main/handler.py`). It applies the blur-fusion foreground estimator of Forte (2021) to byte RGB and a byte-quantized alpha mask, first with radius 30 and then with radius 6, returning byte sRGB foreground. The output retains each method's original continuous alpha. B0 keeps the observed RGB with BiRefNet alpha, B1 uses the refined color with BiRefNet alpha, B2 uses the same color routine with AlphaRAE alpha, and Ours uses the joint decoder. B0/B1 share identical alpha, as do B2/Ours. The latter pair isolates color reconstruction: the AIM means in Table 4 favor Ours on foreground RGB L1, nine-background composite MSE, and black/white composite LPIPS.

Deployed runtime. Timing uses batch size one, five warmup calls, and three passes over 48 fixed native conditions, with CUDA synchronization around each call on one A800. The common RGB DiT and RGB decoder, model loading, host-to-device transfer, image loading, and PNG export are excluded. The adapter uses BF16 and its RGBA decoder FP32. BiRefNet uses FP32 at its standard 1024-pixel internal resolution before its alpha is resized to 256. The latency ratio is therefore specific to these deployed settings, rather than a matched-precision or full-pipeline speedup.

Table 18: Incremental RGBA output cost after shared RGB generation. Bold marks lower measured cost; the two paths use different precision and internal image size. Memory is peak allocated GPU memory during the measured path. All 144 case/repeat records per method are retained.

Path	Mean ms	Median ms	p90 ms	GiB
BiRefNet FP32	207.50	207.49	207.61	3.139
Adapter BF16 + decoder FP32	**34.59**	**34.37**	**35.00**	**1.742**

The adapter has 36.34M parameters, and its incremental path also requires the 415.65M RGB decoder and 0.264M spatial readout, totaling 452.25M active parameters. BiRefNet has 220.18M parameters. Real RGB observations additionally require the 303.15M frozen DINO encoder. The local paired path including this encoder averages 53.7 ms for AlphaRAE; the earlier same-protocol teacher measurement is 208.3 ms. Teacher predictions and local paired metrics were reused after checking the deterministic observation protocol and saved RGB previews. Teacher timing is historical, not from the current adapter measurement session; it should not be interpreted as a contemporary paired speedup.

I Matched-Budget Reconstruction and Stability

Independent-alpha control. The RGB RAE plus independent-alpha baseline uses the same K23 RGB initializer, 7,779 training and 422 validation identities, 60/20/20 input mixture, five loss coefficients, global batch eight, and 50,000-update schedule as Full. It retains 1024 semantic channels and appends a 64-channel alpha code on the 64×64 grid. The RGB decoder reads the semantic code; an independent alpha readout consumes the alpha code. Its 420.000M trainable parameters are 0.087% below Full's 420.364M. This preserves nominal latent shape while testing whether joint color–opacity detail adds value beyond a separate alpha carrier.

Table 19: Budget-matched K23 reconstruction on local TEST434. All alpha errors are multiplied by 10^3. Equal latent scalars do not imply equal information content or compressed bit rate.

Method	Latent scalars	Trainable (M)	PSNR ↑	LPIPS ↓	Alpha L1 ↓
Full	4,456,448	420.364	38.831	0.01463	1.737
Independent alpha	4,456,448	420.000	38.230	0.01740	2.020

Capacity curve. We vary only the learned detail suffix while retaining the K23 semantic code and evaluation protocol. The comparison uses suffix widths $d \in \{16, 32, 64\}$; scalar counts are $(1024 + d)64^2$. Table 20 reports the resulting rate–quality trend in latent scalar counts, not entropy-coded bits or rate matching to AlphaVAE.

Table 20: K23 detail-capacity comparison on TEST434 under the same evaluation protocol.

d	Latent scalars	Trainable (M)	PSNR ↑	LPIPS ↓	Alpha L1 ↓
16	4,259,840	420.344	37.900	0.01840	2.300
32	4,325,376	420.351	38.470	0.01615	1.930
64	4,456,448	420.364	38.831	0.01463	1.737

Shared-readout stability. On the 422-image RGBA validation cohort, the displayed Full and separate-head endpoints are 38.488 and 38.080 dB ($+0.408$ dB). Three additional paired runs give 38.524 versus 38.167, 38.423 versus 38.081, and 38.499 versus 38.102 dB, corresponding to Full-minus-separate-head differences of $+0.357$, $+0.342$, and $+0.397$ dB. Across all four pairs, the mean difference and sample standard deviation are $+0.376 \pm 0.032$ dB. The independent-alpha comparison below tests a different control.

Training-seed variation. Full and the independent-alpha control are evaluated with three fixed seeds each. Their paired PSNR difference is $+0.60 \pm 0.12$ dB across seeds (mean and sample standard deviation). A two-sided Student-t interval computed from these aggregates is approximately $[+0.30, +0.90]$ dB at 95% confidence, assuming independent, approximately normal seed-paired differences; it is necessarily uncertain with only three seeds. Table 21 reports the per-method means and standard deviations. Image-level intervals condition on each checkpoint and do not replace training-seed variation.

Table 21: Three-seed K23 stability on local TEST434. Values are mean ± sample standard deviation across checkpoints; the last row is the paired Full-minus-control difference. Alpha errors are multiplied by 10^3.

Method	PSNR ↑	LPIPS ↓	Alpha L1 ↓
Full, three seeds	38.800 ± 0.052	0.01471 ± 0.00009	1.736 ± 0.015
Independent alpha, three seeds	38.200 ± 0.080	0.01740 ± 0.00025	2.030 ± 0.055
Full minus independent	$+0.60 \pm 0.12$	-0.00269 ± 0.00020	-0.294 ± 0.052

J GENERATED-IMAGE PREFERENCE

The same 64 fixed RGB conditions support separate blinded B1-versus-Ours and B2-versus-Ours comparisons with 53 raters split into three groups of roughly equal size. B2 holds AlphaRAE's alpha fixed, so this second comparison probes whether the joint foreground-color readout improves perceived quality. The groups report approximately $(53, 50, 49)$ mean wins per 64 pairs against B1 and $(47, 49, 48)$ against B2. Averaging the three group means yields approximately 79% and 75% preference, respectively. Cases were fixed before rating; left/right order was randomized per rater, and each pair required a binary choice. The group variation describes rater subsets, not uncertainty across image conditions.

Table 22: Frozen same-RGB human preference on 64 image pairs, evaluated by 53 raters in three groups. Group means and derived percentages are approximate.

Reference	Raters	Pairs	Group mean wins ↑	Ours preference ↑
B1, refined BiRefNet	53	64	$(53, 50, 49)$	$\simeq 79\%$
B2, same-alpha refinement	53	64	$(47, 49, 48)$	$\simeq 75\%$

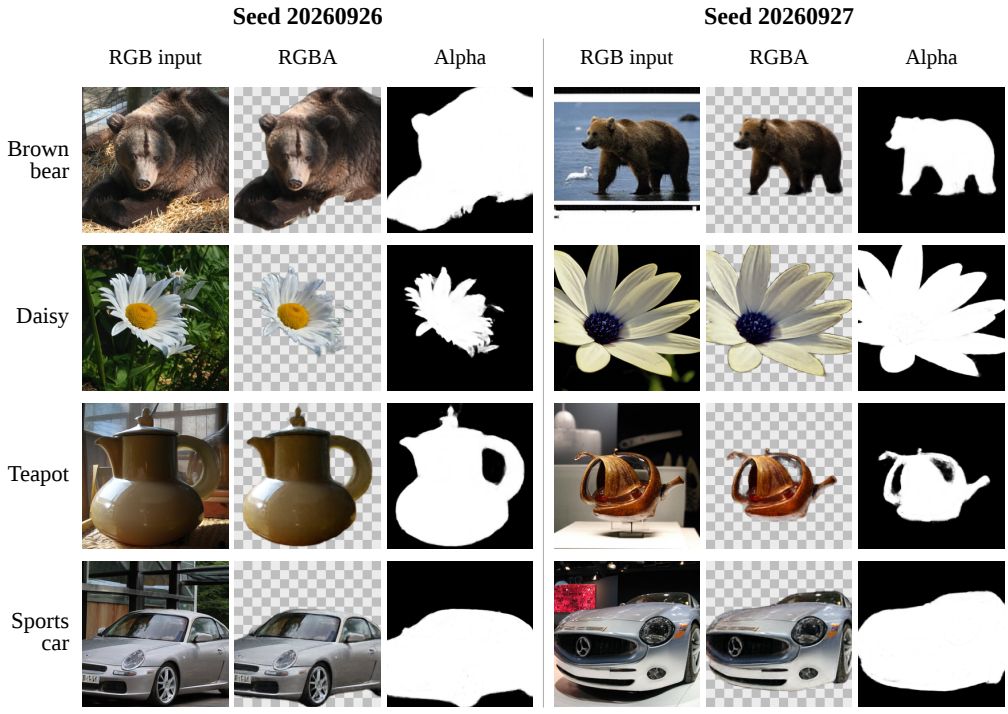

Figure 6: All eight previously fixed end-to-end class/seed examples, evaluated with the final adapter. RGB, joint-decoded RGBA, and predicted alpha are shown. These four classes are represented in pseudo augmentation; source RGB pixels match the previous checkpoint's conditions.

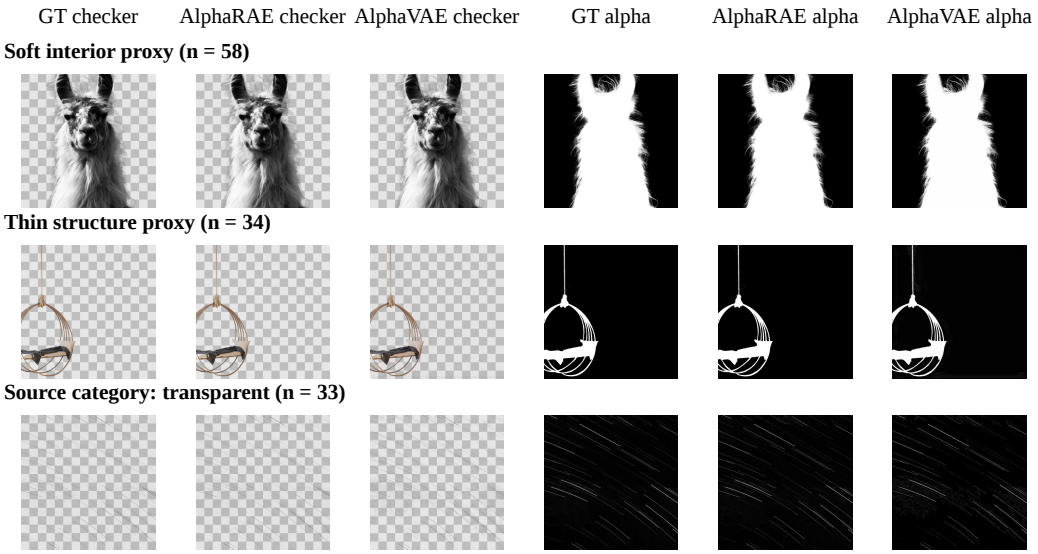

Figure 7: K23 reconstruction of the first manifest-ordered distinct external examples in the soft-interior proxy, thin-support proxy, and source *transparent* category. Images are not selected by model score. Both model columns use the same FP32 CUDA preprocessing. Direct alpha is shown without contrast amplification. Geometry and source labels do not imply that the examples represent glass or liquid.