When Local Rules Create Global Order: Self-Organized Representation Learning for Latent Diffusion Models

Supplementary Material

A. Derivation of Structural Uniformity

In this section, the derivation of the *Remark* is provided. Under the joint constraints of the remapping, smoothness, and dispersity rules, the system converges to a configuration where the minimal separation distances between centroids are equalized.

Let $\mathcal{Z} \subset \mathbb{R}^D$ denote the bounded latent space of dimension D. We consider a set of K learnable centroids $\mathcal{C} = \{\hat{\mathbf{z}}_1, \ldots, \hat{\mathbf{z}}_K\}$. The remapping and smoothness rule collectively ensure that the learnable centroids faithfully represent the local data distribution within their respective Voronoi regions, thus guaranteeing the semantic and structural fidelity of their surrounding elements. Then the dispersity rule aims to maximize the minimal separation between these centroids. We define the separation radius r_i for each centroid $\hat{\mathbf{z}}_i$ such that the characteristic distance to its nearest neighbor is $d_i = 2r_i$. And we approximate the volume occupied by the i-th centroid as proportional to d_i^D.

Assuming the latent space volume is $|\mathcal{Z}|$, the volume constraint can be expressed as $\sum_{i=1}^{K} c \cdot d_i^D \leq |\mathcal{Z}|$, where c is a geometric constant depending on the metric and dimension. For simplicity, we absorb constants into the bound $V_{max} = |\mathcal{Z}|/c$.

Without loss of generality, assume indices are ordered such that d_1 is the minimum separation. The optimization problem can be formally stated as:

$$\begin{cases} arg\min \quad -d_1 \\ s.t. \quad g_1(d_1, d_2, ..., d_K) = \sum_{i=1}^{K} d_i^D - V_{max} \leq 0 \\ \qquad g_2(d_1, d_2, ..., d_K) = d_1 - d_2 \leq 0 \\ \qquad g_3(d_1, d_2, ..., d_K) = d_1 - d_3 \leq 0 \\ \qquad\qquad \vdots \\ \qquad g_K(d_1, d_2, ..., d_K) = d_K - d_2 \leq 0 \end{cases} \tag{3}$$

The constraint g_1 enforces the boundary of the latent space, while constraints $g_2, \ldots, g_K$ enforce that d_1 remains the minimum or equal to the minimum among all distances.

To analyze the properties of the solution space, we evaluate the gradients of the active constraints at the optimal solution $\boldsymbol{d}^* = (d_1^*, d_2^*, \cdots, d_K^*)$. The set of gradient vectors is derived component-wise as follows:

$$\begin{aligned} \nabla g_1(\boldsymbol{d}^*) &= \left(Dd_1^{D-1}, Dd_2^{D-1}, \ldots, Dd_K^{D-1}\right) \\ \nabla g_2(\boldsymbol{d}^*) &= \left(\frac{\partial g_2}{\partial d_1}, \frac{\partial g_2}{\partial d_2}, \ldots, \frac{\partial g_2}{\partial d_K}\right) = (1, -1, 0, 0, \ldots, 0) \\ \nabla g_3(\boldsymbol{d}^*) &= \left(\frac{\partial g_3}{\partial d_1}, \frac{\partial g_3}{\partial d_2}, \ldots, \frac{\partial g_3}{\partial d_K}\right) = (1, 0, -1, 0, \ldots, 0) \\ &\vdots \\ \nabla g_K(\boldsymbol{d}^*) &= \left(\frac{\partial g_K}{\partial d_1}, \frac{\partial g_K}{\partial d_2}, \ldots, \frac{\partial g_K}{\partial d_K}\right) = (1, 0, 0, 0, \ldots, -1) \end{aligned} \tag{4}$$

Observing the gradient structures, it is evident that the vectors $\{\nabla g_i(\boldsymbol{d}^*)\}_{i=2}^{K}$ are linearly independent. Furthermore, given that the separation distances are strictly positive ($d_1, \ldots, d_K > 0$), the gradient $\nabla g_1(\boldsymbol{d}^*)$ contains strictly positive components and cannot be expressed as a linear combination of the sparse vectors $\{\nabla g_i(\boldsymbol{d}^*)\}_{i=2}^{K}$. Consequently, the optimization problem satisfies the Linear Independence Constraint Qualification (LICQ).

Thus, we can proceed by applying the method of Lagrange multipliers. The Lagrangian function associated with Equation 3 is formulated as:

$$L(\boldsymbol{d}, \mu_1, \ldots, \mu_K) = -d_1 + \sum_{i=1}^{K} \mu_i g_i(\boldsymbol{d}) \tag{5}$$

where $\boldsymbol{d} = (d_1, \ldots, d_K)$ and μ_i represents the Lagrange multiplier for the i-th constraint. The first-order necessary conditions for optimality, known as the Karush-Kuhn-Tucker (KKT) conditions, are given by:

$$\begin{cases} \nabla_{\boldsymbol{d}} L = 0 \\ \mu_i g_i(\boldsymbol{d}) = 0, & i = 1, \ldots, K \\ \mu_i \geq 0, & i = 1, \ldots, K \\ g_i(\boldsymbol{d}) \leq 0, & i = 1, \ldots, K \end{cases} \tag{6}$$

Analyzing these conditions yields a symmetric optimal solution:

$$\begin{cases} d_1 = d_2 = \cdots = d_K \\ g_1(\boldsymbol{d}) = 0 \end{cases} \tag{7}$$

This solution implies that the total volume is maximized when the separation distances d_i for all centroids are equal. Geometrically, this leads to a uniform distribution of representations, exemplified by a honeycomb structure in a two-dimensional plane, where equidistant centroids form a stable lattice that maximizes spatial efficiency.

Table 10. Detailed training hyperparameters for models on the CelebA-HQ dataset.

	VAE	WAE	RV-VAE	EQ-VAE	VA-VAE	Ours
f	4	4	4	4	4	4
z-shape	$64 \times 64 \times 3$	$64 \times 64 \times 3$	$64 \times 64 \times 3$	$64 \times 64 \times 3$	$64 \times 64 \times 3$	$64 \times 64 \times 3$
K	-	-	-	-	-	8192
Diffusion steps	1000	1000	1000	1000	1000	1000
Channels	224	224	224	224	224	224
Depth	2	2	2	2	2	2
Channel multiplier	1,2,3,4	1,2,3,4	1,2,3,4	1,2,3,4	1,2,3,4	1,2,3,4
Attention resolutions	32, 16, 8	32, 16, 8	32, 16, 8	32, 16, 8	32, 16, 8	32, 16, 8
Head channels	32	32	32	32	32	32
Learning rate	4.5e-6	4.5e-6	4.5e-6	4.5e-6	4.5e-6	4.5e-6

Table 11. The efficiency analysis comparison on ADE20K.

Model	#Param	FLOPS
VAE	72.8M	473.61G
WAE	72.8M	473.55G
RV-VAE	72.8M	473.61G
EQ-VAE	72.8M	473.61G
VA-VAE	72.8M	473.61G
Ours	72.8M	473.61G

Table 12. The details about batch, epoch and training time set in the training process.

Task	Dataset	Batch	Epoch	Training time
Reconstruction	ADE20K	40	80	24h
Reconstruction	Celeb-HQ	40	80	24h
Reconstruction	FFHQ	40	40	26h
Reconstruction	LSUN-Churches	40	20	22h
Generation	Celeb-HQ	120	200	24h

B. Efficiency analysis

Model complexity comparison. A precise comparison of model complexity is detailed in Table 11, which systematically reports the number of parameters (#Param) and the Floating Point Operations per second (FLOPS) required by each architecture. Notably, all evaluated models—VAE, WAE, RV-VAE, EQ-VAE, VA-VAE, and Ours—are designed to possess an almost identical parameter count (approximately 72.8 Million) and computational cost (approximately 473.6 GFLOPS). This standardization ensures that performance differences are attributable to architectural innovations rather than mere variations in model capacity. Furthermore, Table 10 lists the specific hyperparameters utilized for the training of VAE, WAE, RV-VAE, EQ-VAE, VA-VAE, and the our proposed method.

The K on computation overhead. The additional cost scales linearly with centroid count K: $\mathcal{O}(N \times K)$, where N is the batch size. In practice, K remains moderate and the overhead is small compared to the overall autoencoder and diffusion computation. For example, with $K = 8192$, SORL introduces only an additional 67.15M FLOPs, which is negligible relative to the total training cost 473.61G FLOPs.

Training time. With almost the same parameter size and FLOPS, VAE, WAE, RV-VAE, EQ-VAE, VA-VAE and Ours require almost the same training hours, as shown in Table 12.

C. More Visualization Results

C.1. More generation results

To further validate the efficacy of the proposed model, we present additional unconditional generation results on the CelebA-HQ dataset in Fig. 7. This figure provides a visual comparison of synthesized images generated by ours against several baseline models, specifically VAE, RV-VAE, EQ-VAE, and VA-VAE. The results clearly demonstrate the superior capability of our approach in capturing intricate facial details and generating diverse, high-fidelity, and realistic facial structures compared to existing variational autoencoder variants.

C.2. More cross-domain results

Additional cross-domain visualization comparisons are presented in Fig. 8, Fig. 9 and Fig. 10, covering the DIV2K, LSDIR, and MS-COCO datasets. Furthermore, multi-resolution results specifically for the DIV2K dataset are provided in Fig. 11 and Fig. 12. In the comparison utilizing cross-domain datasets, our proposed model consistently outperforms VAE, WAE, RV-VAE, EQ-VAE and VA-VAE in reconstructing diverse elements, including animals, architecture, text, and landscapes. Furthermore, when evaluating cross-domain reconstruction across multi-resolution settings, our model maintains a visibly superior visualization quality compared to the other five competing architectures. These compelling results underscore the effective-

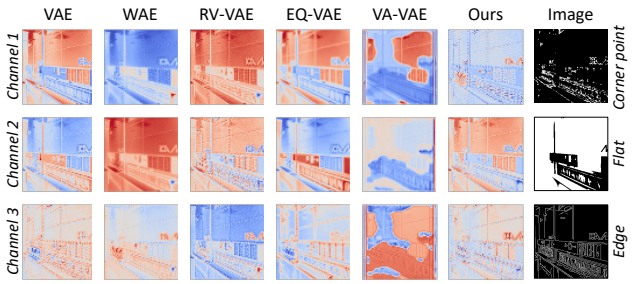

Figure 6. Visualization of latent channel disentanglement. We use DC-AE 1.5 [2] to visualize the first three latent channels from each model, compared with reference geometric features (Corner Point, Flat, Edge) in the rightmost column. Under SORL, Channel 1 aligns with corner structures, Channel 2 focuses on flat regions, and Channel 3 corresponds to edge patterns, revealing a stable and interpretable separation of image primitives. In contrast, other models produce mixed or overlapping activations, indicating weaker disentanglement and less organized latent representations.

ness and generalization capability of our underlying latent representation structure in handling both cross-domain and multi-resolution reconstruction tasks.

C.3. Latent Feature Disentanglement.

In addition to improving smoothness and dispersity in the latent space, our SORL framework also promotes functional disentanglement across latent channels. To study this effect, we visualize the activation maps of individual channels and compare them with VAE-based baselines. As shown in Fig. 6, baseline models typically exhibit considerable redundancy, with multiple channels responding to similar spatial regions or capturing mixed semantic cues. In contrast, SORL produces channels that specialize in distinct geometric primitives such as flat areas, edges, and corner points. This emergent separation suggests that self-organization reduces representational overlap and yields a more structured, interpretable latent space, where different channels capture complementary aspects of the image.

C.4. Further Details on Generative Modeling

We follow the standard two-stage latent diffusion pipeline. Specifically, we first train each autoencoder (e.g., SORL) to obtain its latent representation, and then train a diffusion model from scratch on the fixed latent codes produced by the frozen encoder, The diffusion architecture and training protocol are kept identical across all methods, so performance differences can be attributed solely to the quality of the learned latent space. No pre-trained diffusion model is fine-tuned.

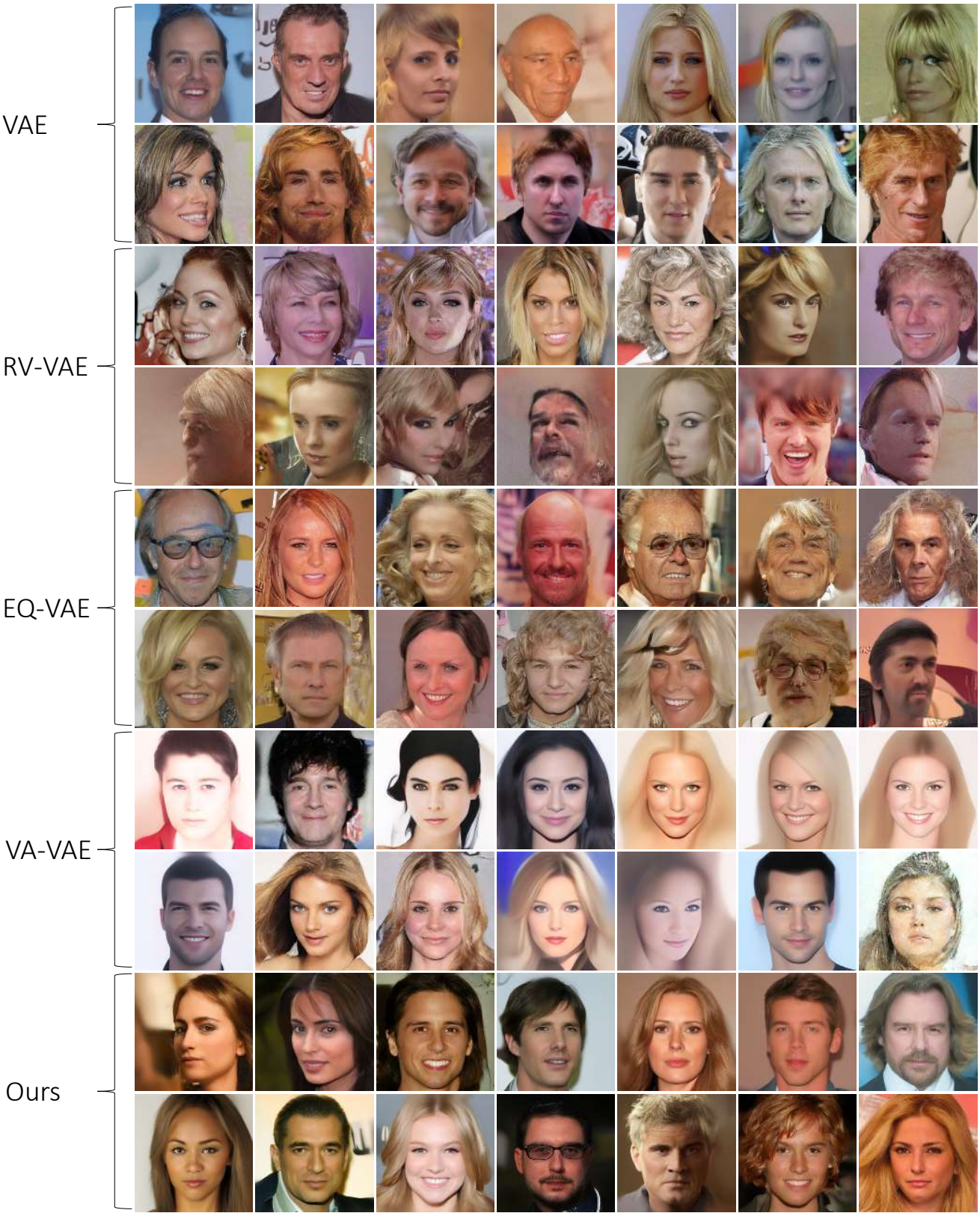

Figure 7. Unconditional generation results on CelebA-HQ dataset.

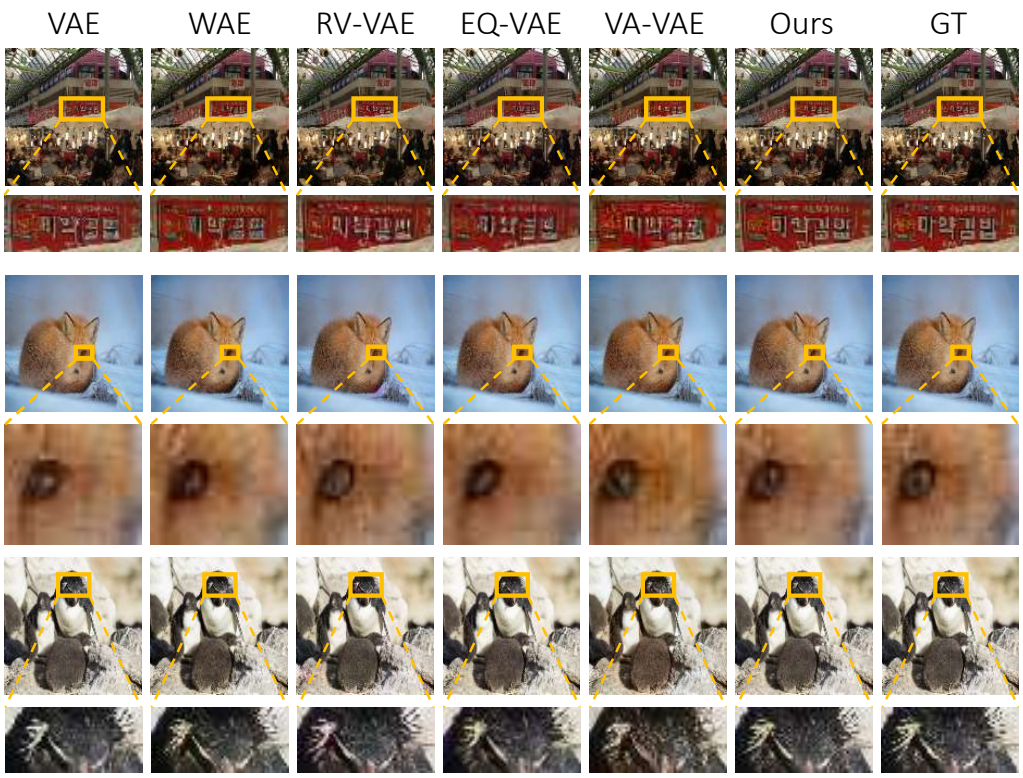

Figure 8. Cross-domain reconstruction results on DIV2K dataset.

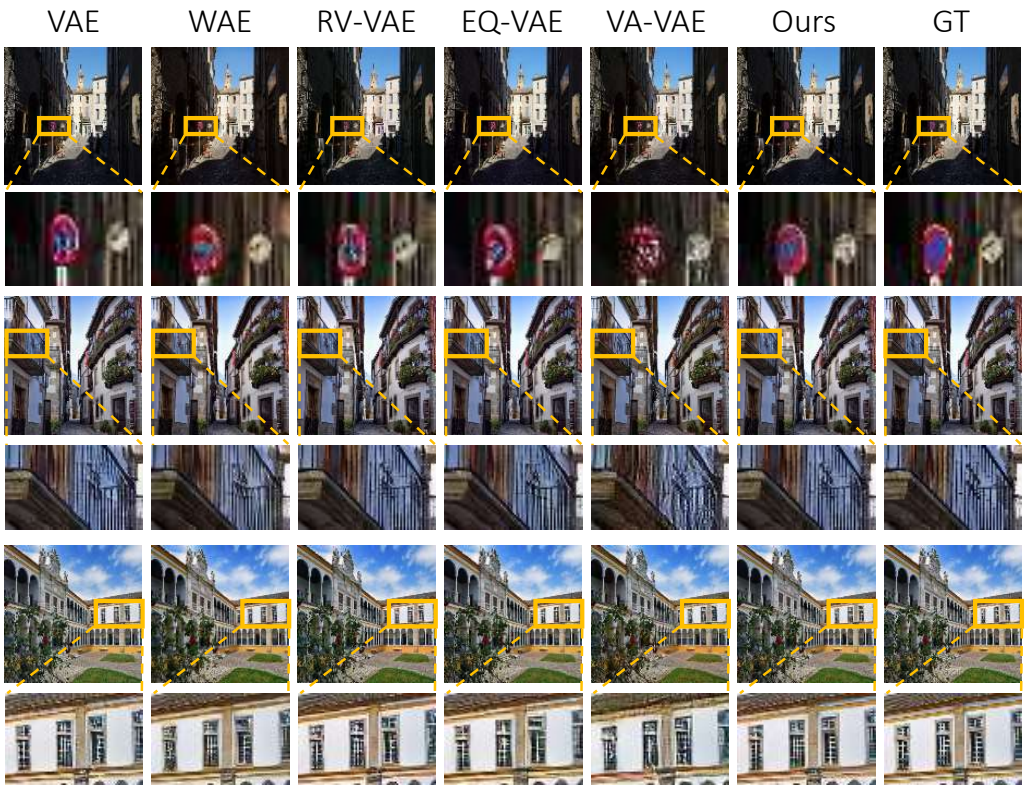

Figure 9. Cross-domain reconstruction results on LSDIR dataset.

Figure 10. Cross-domain reconstruction results on MS-COCO dataset.

Figure 11. Cross-resolution reconstruction on the cross-domain dataset. This figure primarily highlights the reconstruction fidelity of the model when handling high-frequency semantic structures like text regions, validating its ability to maintain detailed information.

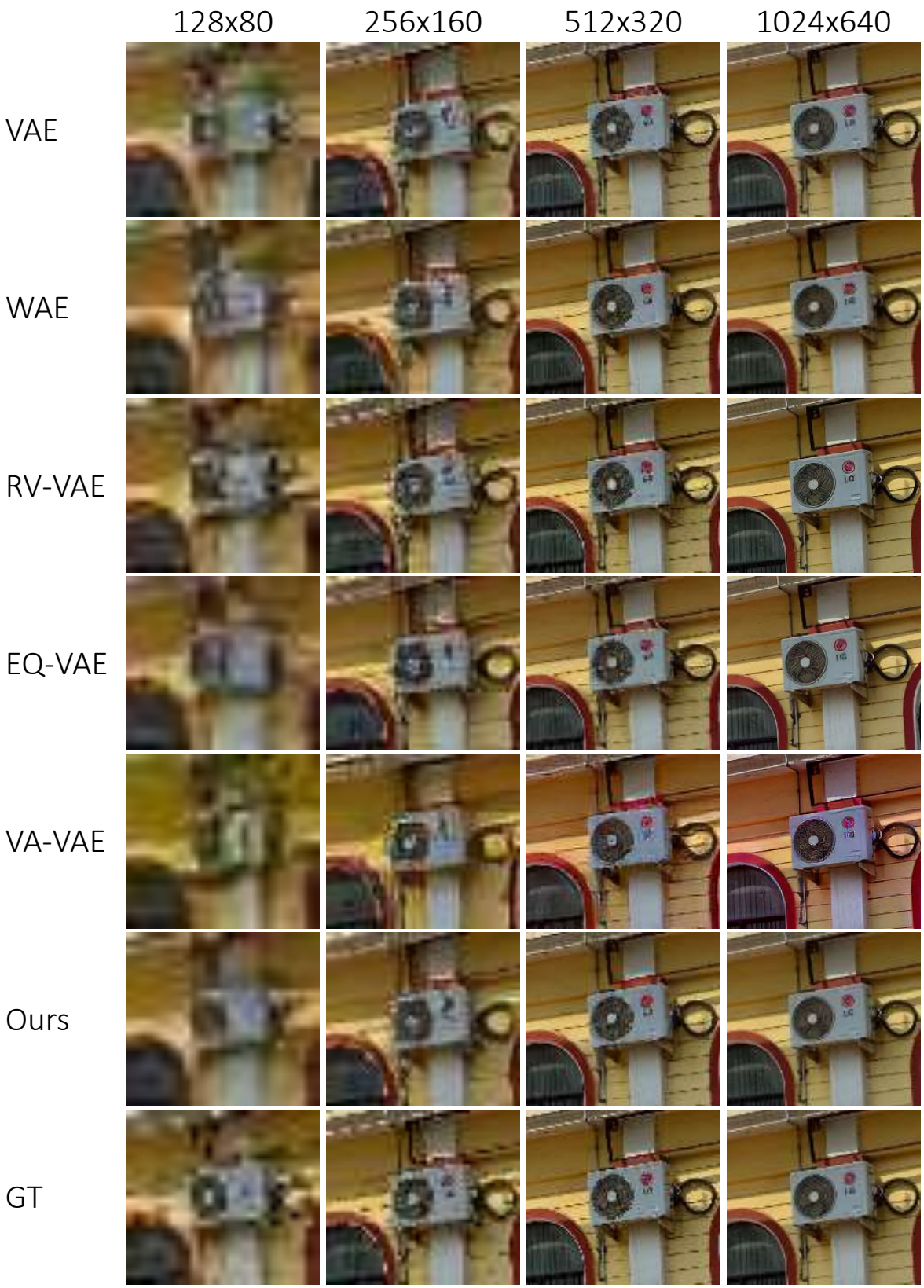

Figure 12. Cross-resolution reconstruction on the cross-domain dataset (Cont.). Serving as a supplement, this figure focuses on demonstrating the model's reconstruction performance on complex texture regions like outdoor air conditioner units to evaluate its capacity for synthesizing fine geometry and material textures.