ZoomLDM: Latent Diffusion Model for multi-scale image generation

Supplementary Material

We organize the supplementary as follows:

S1. ZoomLDM on satellite images

In the main text, we focused on the digital histopathology domain and how our multi-scale diffusion model can prove useful in generation and downstream tasks. However, gigapixel images also concern the remote sensing domain, where satellite images regularly are in the range of 10000×10000 pixels. To show the wide applicability of our multi-scale approach, we trained ZoomLDM on satellite images from the NAIP dataset [13], specifically using NAIP images from the Chesapeake subset of [12]. NAIP images are at 1m resolution – the distance between pixel centers is 1m. We follow the same dataset preparation approach and extract 256×256 patches at four different scales with pixels corresponding to 1m, 2m, 4m, and 8m resolutions. For the SSL encoder, we resort to a pre-trained DINOv2 model [10], which has been known to perform well across many modalities, including satellite.

In Table S1, we provide the per-resolution FID numbers our model achieves. Similarly to histopathology, we observe that training a cross-scale model benefits the scales where there is not enough data to train a single-scale model on (8m resolution in this case). We also showcase patches generated by ZoomLDM at all four resolutions in Figure S9. We present examples from the satellite ZoomLDM variant in S3.2 and S4.3.

In Table S2, we provide the FID numbers for large satellite image generation (1024×1024). Our satellite ZoomLDM model achieves significantly better results on crop FID while achieving similar CLIP FID; this showcases our ability to synthesize high-quality images that simultaneously maintain global consistency.

Resolution	1m	2m	4m	8m
# Training patches	365 k	94 k	25 k	8.7 k
ZoomLDM	**10.93**	**7.77**	**7.34**	**8.46**
SoTA model	11.5 [6]	23.61	37.52	65.45

Table S1. NAIP FID values obtained by ZoomLDM versus training a state-of-the-art diffusion model on a single resolution. Having a shared model across multiple scales improves the generation quality for the data-scarce scales. For resolutions >1m we retrain the model of [6] on the samples from that resolution only.

Method	CLIP FID	Crop FID
Graikos et al. [6]	6.86	43.76
∞-Brush [9]	**6.32**	48.65
ZoomLDM	7.90	**13.25**

Table S2. CLIP and Crop FID values (lower is better) for large (1024×1024) satellite images. ZoomLDM outperforms previous works while also maintaining a reasonable inference time.

S2. Ablation on SSL encoder and Summarizer

We retrain ZoomLDM with (i) a weaker SSL encoder (HIPT [1]) and (ii) both a weaker SSL encoder and a simpler summarizer network (CNN vs ViT). Table S3 shows that replacing UNI with HIPT degrades performance and further replacing the ViT summarizer network with a simple 4-layer CNN leads to a greater decline.

When comparing the downstream performance of the denoiser features on a multiple-instance learning task (MIL) we also see a decrease in performance when using a 'weaker' conditioning encoder. We believe that training a diffusion model

SSL	Summarizer	FID across magnifications ↓								MIL (AUC) ↑	
		20×	10×	5×	2.5×	1.25×	0.625×	0.3125×	0.15625×	Subtyping	HRD
HIPT [1]	CNN	18.88	16.75	19.31	16.01	14.45	14.21	15.44	18.47	86.20	72.44
HIPT [1]	ViT	13.49	14.42	15.84	13.32	14.32	12.31	16.25	19.90	87.26	75.92
UNI [2]	ViT	**6.77**	**7.60**	**7.98**	**10.73**	**8.74**	**7.99**	**8.34**	**13.42**	**94.49**	**85.25**

Table S3. Ablation on SSL encoder and summarizer network architecture. Using a weaker SSL encoder or summarizer leads to worse performance in both generation and downstream discriminative tasks.

conditioned on SSL representations complements the discriminative SSL pre-training with the newly learned generative features. In all our experiments, improved image quality leads to better downstream task performance. Additionally, the SSL encoders used in MIL are usually trained on a single magnification, making our approach a potential way to fuse features across different scales effectively.

S3. Experiment details

S3.1. Summarizer-CDM training details

Summarizer: We train the Summarizer jointly with the LDM. The Summarizer processes the SSL embeddings extracted alongside the image patches and projects them to a latent space that is shared across all scales (cross-magnification latent space). By training jointly with the LDM the Summarizer learns to compress the SSL embeddings into a representation useful for making images.

We pre-process the SSL embedding matrices via element-wise normalization. The Summarizer receives 64 SSL embeddings (or fewer SSL embeddings with appropriate padding to 64 tokens) concatenated with a learned magnification embedding as input. The network consists of a 12-layer Transformer encoder with a hidden dimension of 512, followed by a LayerNorm operation to normalize the output. The 65×512 dimensional output is then fed to the U-Net denoiser via cross-attention.

CDM: To avoid reliance on real images to extract the SSL embeddings required for sampling, we train a Conditioning Diffusion Model (CDM). The CDM is trained to draw samples from the learned cross-magnification latent space. After training the LDM and Summarizer jointly, we train the CDM with the denoising objective to sample from the 65×512 output. See Figure S1 for an overview of the Summarizer and CDM.

We implement the CDM as a Diffusion Transformer [11]. We use the DiT-Base architecture, consisting of 12 layers and a hidden size of 768. We use an MLP to project the output back to the exact channel dimensions as the input. We use a constant learning rate of 10^{-4}, following the implementation of [11]. We present samples generated by the CDM in Figure S8.

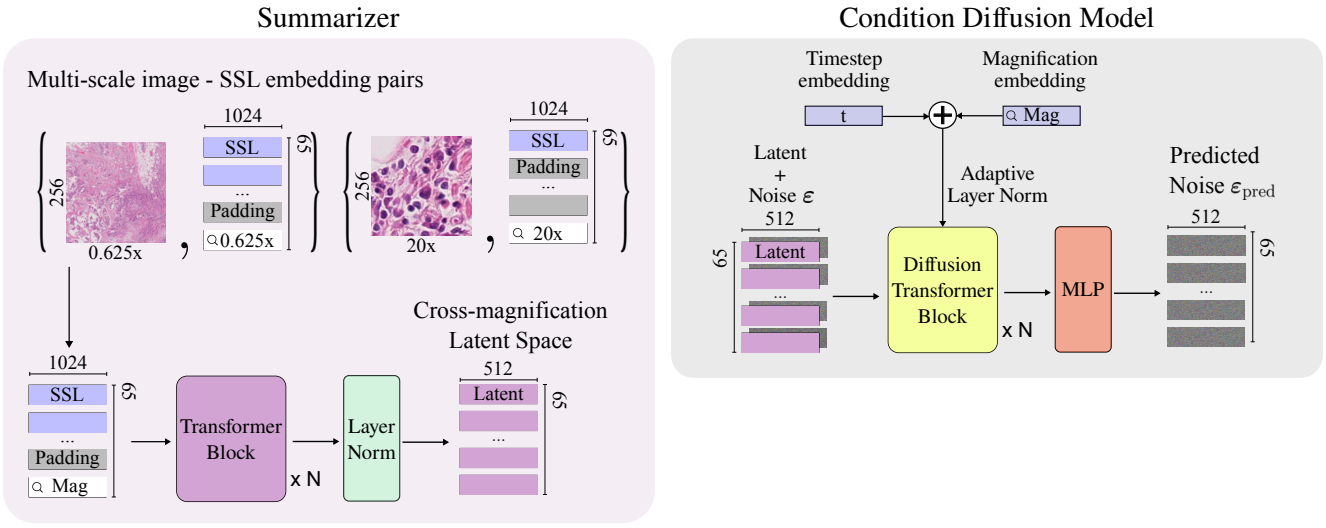

Figure S1. Overview of the Summarizer and Condition Diffusion Model.

S3.2. Joint Sampling

In this section, we present an overview of the joint sampling algorithm. By jointly generating an image that depicts the global context and images that produce local details we are able to synthesize large images at the highest resolution that maintain global coherency. We achieve that by simultaneously generating patches i with high-resolution details $\boldsymbol{x}^i = Dec(\boldsymbol{z}^i)$ and a lower-resolution context $\boldsymbol{x}^L = Dec(\boldsymbol{z}^L)$ that globally guides the structure of the patches.

Our joint sampling method is based on a recent fast sampling algorithm for diffusion models under linear constraints, presented in [5]. The full algorithm is shown in Algorithm S1. We make two key changes to the inference algorithm to perform joint multi-scale sampling: (i) We replace the constraint $\boldsymbol{y}$ with the current estimate of the lower-scale image $Dec(\hat{\boldsymbol{z}}_0^L)$ and (ii) we replace the expensive backpropagation step required in computing the error $\boldsymbol{e}$ with a less memory-intensive approximation using forward passes through the encoder and the decoder.

Utilizing intermediate steps Instead of having access to a measurement $\boldsymbol{y}$ we only have access to the current estimate of the context image. That image is in practice a subsampled version of the spatially arranged patches $\boldsymbol{x}^i$. To relate the two, we rearrange $\boldsymbol{x}_i$ and apply a linear subsampling operator $\boldsymbol{A}$, such as bicubic interpolation. This operator is used to compute the difference between the current synthesized patches and the current context and will be used to update the content of the patch images.

Avoiding backpropagation For latent diffusion models, the original algorithm relies on computing the difference between the context and the patches which it then backpropagates through the decoder to get the direction towards which this error is minimized. However, when we synthesize 4k images, we end up with 256 high-resolution patches, and backpropagating becomes prohibitively memory-intensive. To that end, we propose a modification to the sampling algorithm that replaces the backpropagation step with forward passes through the encoder and decoder.

To produce the high-resolution images, we want to sample z_t under the guidance of the lower-scale image, minimizing a constraint $C(\boldsymbol{z}_t) = ||\boldsymbol{A}Dec(\hat{\boldsymbol{z}}_0(\boldsymbol{z}_t)) - Dec(\hat{\boldsymbol{z}}_0^L)||_2^2$. Algorithm S1 requires us to compute the direction $\boldsymbol{e}$ of $\hat{\boldsymbol{z}}_0$ towards which the constraint C is minimized and uses it to update the current diffusion latent as

$$\boldsymbol{g} = \frac{\hat{\boldsymbol{z}}_0(\boldsymbol{z}_t + \delta \boldsymbol{e}) - \hat{\boldsymbol{z}}_0(\boldsymbol{z}_t)}{\delta} \tag{S1}$$

$$z_t' = z_t + \lambda g. \tag{S2}$$

However, to calculate $\boldsymbol{g}$ we need $\boldsymbol{e} = \frac{\partial C}{\partial \hat{\boldsymbol{z}}_0}$ which we can calculate by backpropagating through the decoder model. Since this is computationally burdensome, we apply the chain rule to get

$$\boldsymbol{e} = \frac{\partial C}{\partial \hat{\boldsymbol{z}}_0} = \left(\frac{\partial Dec(\hat{\boldsymbol{z}}_0)}{\partial \hat{\boldsymbol{z}}_0}\right)^T \frac{\partial C}{\partial Dec(\hat{\boldsymbol{z}}_0)} = \left(\frac{\partial Dec(\hat{\boldsymbol{z}}_0)}{\partial \hat{\boldsymbol{z}}_0}\right)^T \boldsymbol{e}_{img}, \quad \boldsymbol{e}_{img} = \boldsymbol{A}^T(\boldsymbol{A}Dec(\hat{\boldsymbol{z}}_0(\boldsymbol{z}_t)) - Dec(\hat{\boldsymbol{z}}_0^L)) \tag{S3}$$

The LDM VAEs that we use (VQ-VAE or KL-VAE) are trained in a way that forces the Jacobian of the Decoder to be approximately orthogonal, through vector quantization or minimizing the KL divergence between the predicted posterior and an isotropic Gaussian. For orthogonal Jacobians Eq. S3 can be simplified into:

$$\boldsymbol{e} = \left(\frac{\partial Dec(\hat{\boldsymbol{z}}_0)}{\partial \hat{\boldsymbol{z}}_0}\right)^T e_{img} \approx \frac{\partial \hat{\boldsymbol{z}}_0}{\partial Dec(\hat{\boldsymbol{z}}_0)} e_{img} \tag{S4}$$

and assuming that the VAE has learned to reconstruct images perfectly, it can be written as:

$$\boldsymbol{e} \approx \frac{\partial \hat{\boldsymbol{z}}_0}{\partial Dec(\hat{\boldsymbol{z}}_0)} e_{img} \approx \frac{\partial Enc(Dec(\hat{\boldsymbol{z}}_0))}{\partial Dec(\hat{\boldsymbol{z}}_0)} e_{img}. \tag{S5}$$

We can now approximate $\boldsymbol{e}$ using finite differences:

$$\boldsymbol{e} \approx \frac{\partial Enc(Dec(\hat{\boldsymbol{z}}_0))}{\partial Dec(\hat{\boldsymbol{z}}_0)} e_{img} \approx \frac{Enc(Dec(\hat{\boldsymbol{z}}_0) + \zeta e_{img}) - Enc(Dec(\hat{\boldsymbol{z}}_0))}{\zeta} \tag{S6}$$

which completely erases the need to perform memory-heavy backpropagation through the decoder model.

A step-by-step description of our joint sampling method can be found in Algorithm S2. We use 50 DDIM steps for our experiments, bicubic upsampling/downsampling for $\boldsymbol{A}$, $\delta = \zeta = 0.005$, $K = 1$, $\lambda = 0.5$. Upon observing noticeable discontinuities along the borders of the high-resolution patches, we apply a simple post-processing step by adding noise and

denoising the patches between, similar to [6]. We provide some results of the joint sampling, visualized in Figures S2,S3 for the histopathology and satellite domains.

Algorithm S1 The algorithm for linear inverse problem solving proposed in [5].

Input: Diffusion model $\hat{\boldsymbol{z}}_0(\boldsymbol{z}_t)$, Enc, Dec, schedule $T_{0,\dots,M}$, subsampling operator $\boldsymbol{A}$, measurement $\boldsymbol{y}$, step size δ, # iterations K, learning rate λ
$\boldsymbol{z}_T \sim N(\boldsymbol{0}, \boldsymbol{I})$
for $t \in \{T_0, T_1, \dots, T_M\}$ **do**
  for $i \in \{1, 2, \dots, K\}$ **do**
    $\boldsymbol{e} = \nabla_{\boldsymbol{z}_0} ||\boldsymbol{A}Dec(\hat{\boldsymbol{z}}_0(\boldsymbol{z}_t)) - \boldsymbol{y}||_2^2$
    $\boldsymbol{g} = \left[\hat{\boldsymbol{z}}_0(\boldsymbol{z}_t + \delta\boldsymbol{e}) - \hat{\boldsymbol{z}}_0(\boldsymbol{z}_t)\right]/\delta$
    $\boldsymbol{z}_t = \boldsymbol{z}_t + \lambda\boldsymbol{g}$
  end for
  $\boldsymbol{z}_t = \text{DDIM}(\boldsymbol{z}_t, \hat{\boldsymbol{x}}_0, s)$
end for
Return: $\boldsymbol{x}_0$

Algorithm S2 The proposed modification to Algorithm S1.

Input: Diffusion model $\hat{\boldsymbol{z}}_0(\boldsymbol{z}_t)$, Enc, Dec, schedule $T_{0,\dots,M}$, subsampling operator $\boldsymbol{A}$, detail scale s, context scale s_L, step sizes δ, ζ, # iterations K, learning rate λ
$\boldsymbol{z}_T \sim N(\boldsymbol{0}, \boldsymbol{I})$
$\boldsymbol{z}_T^L \sim N(\boldsymbol{0}, \boldsymbol{I})$
for $t \in \{T_0, T_1, \dots, T_M\}$ **do**
  for $i \in \{1, 2, \dots, K\}$ **do**
    $\boldsymbol{e}_{img} = \boldsymbol{A}^T(\boldsymbol{A}Dec(\hat{\boldsymbol{z}}_0(\boldsymbol{z}_t)) - Dec(\hat{\boldsymbol{z}}_0^L))$
    $\boldsymbol{e} = \left[Enc(Dec(\hat{\boldsymbol{z}}_0) + \zeta\boldsymbol{e}_{img}) - Enc(Dec(\hat{\boldsymbol{z}}_0))\right]/\zeta$
    $\boldsymbol{g} = \left[\hat{\boldsymbol{z}}_0(\boldsymbol{z}_t + \delta\boldsymbol{e}) - \hat{\boldsymbol{z}}_0(\boldsymbol{z}_t)\right]/\delta$
    $\boldsymbol{z}_t = \boldsymbol{z}_t + \lambda\boldsymbol{g}$
  end for
  $\boldsymbol{z}_t = \text{DDIM}(\boldsymbol{z}_t, \hat{\boldsymbol{x}}_0, s)$
  $\boldsymbol{z}_t^L = \text{DDIM}(\boldsymbol{z}_t^L, \hat{\boldsymbol{x}}_0, s_L)$
end for
Return: $\boldsymbol{x}_0$

S3.3. Image Inversion

In this section, we present our image inversion algorithm, which is crucial for performing the super-resolution task described in the main text. The conditioning we provide to the model is a set of SSL embeddings extracted at the highest resolution available. For instance, in histopathology, the SSL conditions are extracted at $20\times$. Thus, when we are given a single image at any magnification that we want to super-resolve we do not have access to this conditioning and are limited to using the model in an unconditional manner. The unconditional model is available since we randomly drop the conditioning during training, to implement classifier-free guidance [7] during sampling. However, recent works have argued that when using the diffusion model to sample with linear constraints, like super-resolution, conditioning helps in achieving better-fidelity results [3].

Inspired by those findings, we propose a simple algorithm to first *invert* the model and get conditioning for a single image, before super-resolving it. The algorithm is an adaptation of the textual inversion technique of Gal et al. [4], which has seen wide success in text-to-image diffusion models. An overview of the approach is provided in Figure S4.

Given an image $\boldsymbol{I}$ at scale s, we have access to a pre-trained latent denoiser model $\boldsymbol{\epsilon}_\theta(\boldsymbol{z}_t, t, f(\boldsymbol{e}, s))$ where $\boldsymbol{z} = Enc(\boldsymbol{I})$, g is the summarizer model and $\boldsymbol{e}$ are the SSL embeddings that describe the image. We want to draw a sample $\boldsymbol{e}$, that when provided as conditioning to the diffusion model will generate images similar to $\boldsymbol{I}$. From the latent variable perspective of diffusion models, described by Ho et al. [8], we obtain the following lower bound for the log probability of $\boldsymbol{z}$ given a condition $\boldsymbol{e}$

$$\log p(\boldsymbol{z} \mid \boldsymbol{e}) \geq -\sum_{t=1}^{T} w_t(\alpha)\mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{N}(\boldsymbol{0},\boldsymbol{I})}\left[||\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\boldsymbol{z}_t, t, g(\boldsymbol{e}, s)) - \boldsymbol{\epsilon}||_2^2\right], \quad \boldsymbol{z}_t = \sqrt{\alpha_t}\boldsymbol{z} + \sqrt{1-\alpha}\boldsymbol{\epsilon}. \tag{S7}$$

We then employ variational inference to fit an approximate posterior $q(\boldsymbol{e})$ to $p(\boldsymbol{e} \mid \boldsymbol{z})$ from which we want to sample conditions given an input image. We start by defining a lower bound for $\log p(\boldsymbol{z})$

$$\begin{aligned}
\log p(\boldsymbol{z}) &= \log \int_{\boldsymbol{e}} p(\boldsymbol{z}, \boldsymbol{e}) d\boldsymbol{e} = \log \int_{\boldsymbol{e}} q(\boldsymbol{e}) \frac{p(\boldsymbol{z}, \boldsymbol{e})}{q(\boldsymbol{e})} d\boldsymbol{e} \\
&= \log \mathbb{E}_{q(\boldsymbol{e})}\left[\frac{p(\boldsymbol{z}, \boldsymbol{e})}{q(\boldsymbol{e})}\right] \geq \mathbb{E}_{q(\boldsymbol{e})}\left[\log \frac{p(\boldsymbol{z}, \boldsymbol{e})}{q(\boldsymbol{e})}\right] \\
&= \mathbb{E}_{q(\boldsymbol{e})}\left[\log \frac{p(\boldsymbol{z} \mid \boldsymbol{e})p(\boldsymbol{e})}{q(\boldsymbol{e})}\right] = L.
\end{aligned} \tag{S8}$$

By maximizing the bound L w.r.t. the parameters of q we minimize the KL-Divergence between the approximate posterior $q(\boldsymbol{e})$ and the real $p(\boldsymbol{e} \mid \boldsymbol{z})$. We choose a simple Dirac delta $q(\boldsymbol{e}) = \delta(\boldsymbol{e} - \boldsymbol{u})$ as our approximation, which allows us to use the bound from Eq. S7 to simplify the objective

$$\begin{aligned} L &= \mathbb{E}_{q(\boldsymbol{e})}\left[\log p(\boldsymbol{z} \mid \boldsymbol{e}) + \log p(\boldsymbol{e}) - \log q(\boldsymbol{e})\right] = \log p(\boldsymbol{z} \mid \boldsymbol{e} = \boldsymbol{u}) + \log p(\boldsymbol{e} = \boldsymbol{u}) \\ &= -\sum_{t=1}^{T} w_t(\alpha)\mathbb{E}_{\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})}\left[||\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\boldsymbol{z}_t, t, g(\boldsymbol{u}, s)) - \boldsymbol{\epsilon}||_2^2\right] + \log p(\boldsymbol{e} = \boldsymbol{u}). \end{aligned} \tag{S9}$$

Therefore, to draw a sample from the posterior $p(\boldsymbol{e} \mid \boldsymbol{z})$ we optimize Eq. S9 w.r.t. $\boldsymbol{u}$. The result is a single point $\boldsymbol{u}$ that seeks a local mode of $p(\boldsymbol{e} \mid \boldsymbol{z})$.

For the prior term $\log p(\boldsymbol{e})$, we use a simple heuristic, implementing a penalty that maximizes the similarity between the different vectors in the SSL embeddings $\boldsymbol{e}$. This heuristic encourages the model to find embeddings that generate similar patches when used independently. For the denoising terms, we must add random Gaussian noise to the image latent $\boldsymbol{z}$ and denoise at multiple timesteps t. Instead of evaluating multiple timesteps simultaneously, we utilize an annealing schedule that starts from $t = 950$ and linearly decreases to $t = 50$ over the $n = 200$ optimization steps we perform. Overall, the proposed algorithm is similar to textual inversion [4], which utilizes the denoising loss to optimize text tokens $\boldsymbol{t}$.

In Figure S5, we provide qualitative results for our inversion approach. We present two cases, inferring the condition for $5\times$ and $2.5\times$ images. We observe that for $5\times$, which is also the scale used in our super-resolution experiments, our approach can provide conditions that faithfully reconstruct both the $5\times$ image and also give us plausible $20\times$ patches. As we increase

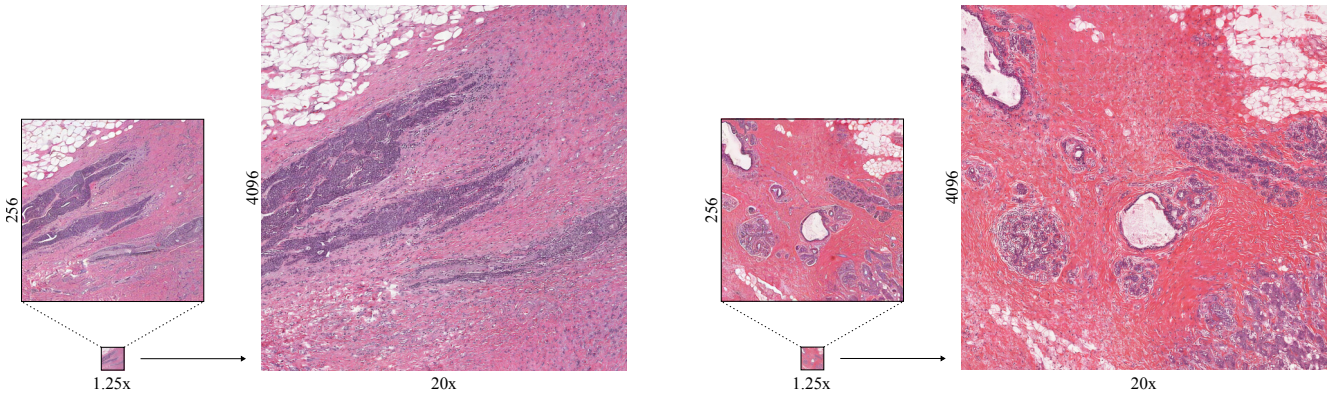

Figure S2. Joint sampling process across two different magnifications for the TCGA-BRCA ZoomLDM model. We jointly generate a 256×256 image at $1.25\times$ and a 4096×4096 image at $20\times$. The $1.25\times$ generation guides the structure of the $20\times$ image by providing the necessary global context that each $20\times$ patch is unaware of. The generated large $20\times$ image has a realistic global arrangement of cells and tissue. Best viewed zoomed-in.

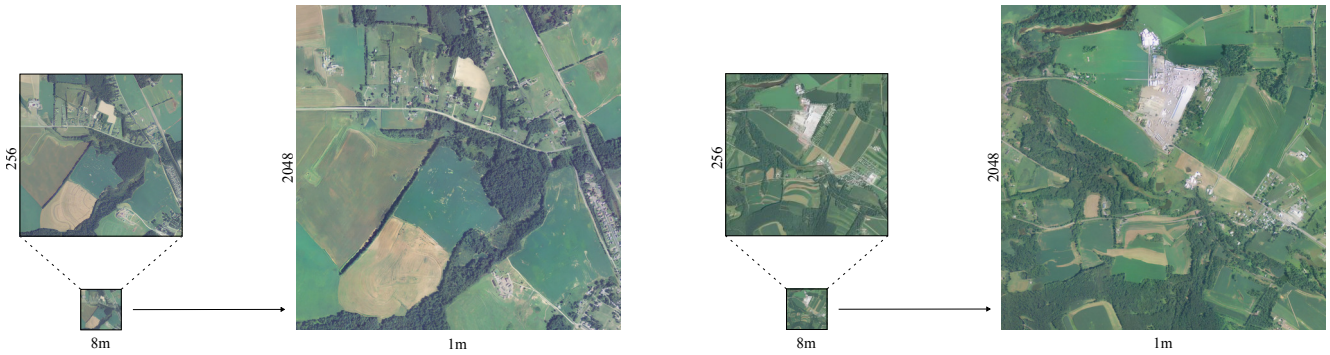

Figure S3. Joint sampling process across two different resolutions for the Satellite ZoomLDM model. We jointly generate a 256×256 image at $8m$ resolution and a 2048×2048 image at $1m$. The $8m$ generation guides the structure of the $1m$ image by providing global coherence, which, otherwise, each $1m$ would be unaware of. The generated large $1m$ image has realistic global structures, with roads and forests neatly arranged across the 2048×2048 canvas. Best viewed zoomed-in.

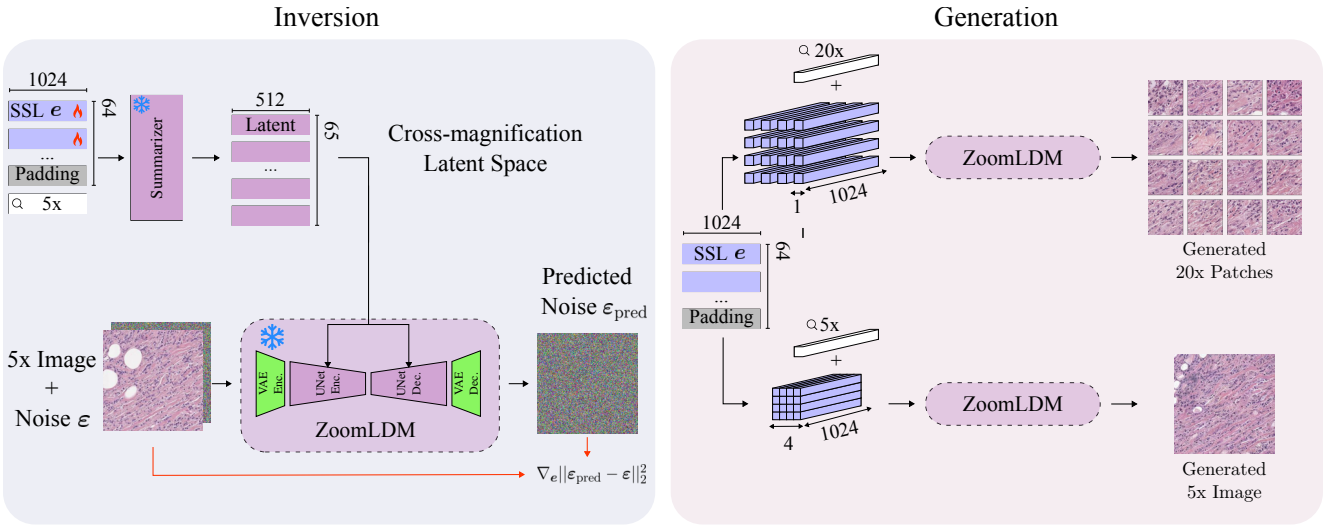

Figure S4. Figure illustrating our pipeline for the image inversion used in the super-resolution task. For a given image we first use the denoising loss to optimize the input, conditioning embeddings. We can then generate variations of the given image and high-resolution patches from it. We use those per-patch embeddings to perform super-resolution, obtaining better results than unconditional super-resolution.

the number of conditions to infer, the $2.5\times$ result remains convincing at the lower scale but struggles to provide reasonable $20\times$ patches. Future work focusing on this inversion approach could provide useful insights into the SSL embeddings used as conditioning, helping understand what they encode and the topology of the latent space created by the SSL encoder.

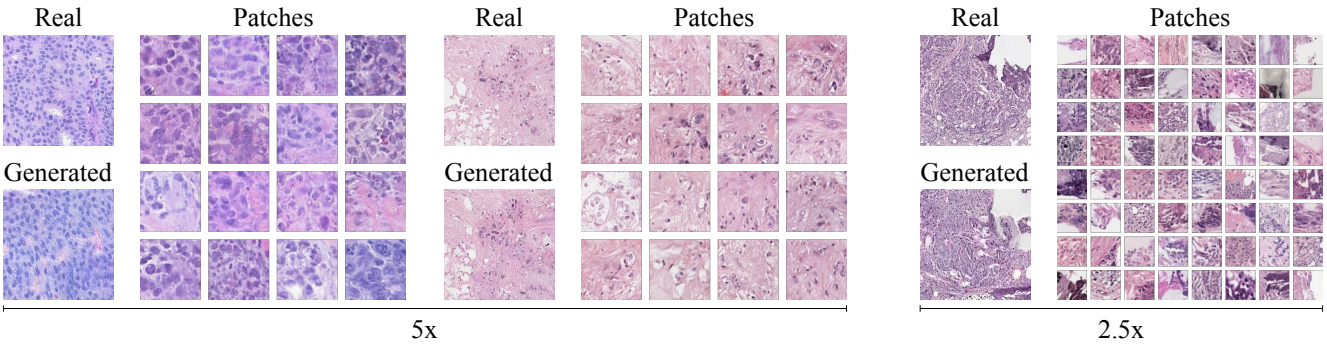

Figure S5. Examples of the image inversion algorithm. Given a real image at any magnification, we infer the SSL embeddings that generated it. We then generate a new, similar-looking image at the same magnification using those embeddings as conditioning. Using the inferred embeddings to generate single patches from the given image yields convincing results at magnifications $> 5\times$.

S4. Additional results

S4.1. More super-resolution baselines

In Tables S4 and S5 we provide additional baselines for the super-resolution task. We use ResShift [15, 16] and StableSR [14] to super-resolve pathology images and compare them to the zero-shot performance of ZoomLDM. Using ZoomLDM in a training-free manner (with condition inference S3.3) remains the best approach for histopathology image super-resolution.

S4.2. Data efficiency and memorization

One of the arguments for training a single model for all scales is that we can learn to generate novel images even at scales with too few samples to learn from. To further demonstrate this, we use our histopathology diffusion model and sample conditions from the Conditioning Diffusion Model (CDM) to generate novel images at $0.15625\times$ magnification. At this scale, both our models have only seen ~ 2500 images and we would expect them to either generate low-quality samples or to

Table S4. Super-resolution results on TCGA-BRCA

Method	SSIM ↑	PSNR ↑	LPIPS↓	CONCH ↑	UNI ↑
ResShift v2 (15 steps) [16]	0.415	19.716	0.431	0.847	0.299
ResShift v3 (4 steps) [15]	0.525	21.528	0.314	0.866	0.311
StableSR no tiling [14]	0.515	21.644	0.315	0.862	0.390
StableSR w/ tiling [14]	0.514	21.618	0.316	0.863	0.388
ZoomLDM (Uncond)	0.591	23.217	0.260	0.936	<u>0.680</u>
ZoomLDM (GT Emb)	0.599	23.273	0.250	<u>0.946</u>	0.672
ZoomLDM (Infer Emb)	<u>0.609</u>	<u>23.407</u>	**0.229**	**0.957**	**0.719**

Table S5. Super-resolution results on BACH

Method	SSIM ↑	PSNR ↑	LPIPS↓	CONCH ↑	UNI ↑
ResShift v2 (15 steps) [16]	0.584	23.256	0.421	0.898	0.621
ResShift v3 (4 steps) [15]	0.751	26.283	0.257	0.898	0.623
StableSR no tiling [14]	0.729	26.203	0.291	0.846	0.547
StableSR w/ tiling [14]	0.729	26.200	0.293	0.845	0.538
ZoomLDM (Uncond)	0.739	29.822	0.235	0.965	0.741
ZoomLDM (GT Emb)	0.732	29.236	0.245	<u>0.974</u>	0.753
ZoomLDM (Infer Emb)	<u>0.779</u>	<u>30.443</u>	**0.173**	**0.974**	<u>0.808</u>

have memorized the training data when using a 400M parameter model in training. Contrary to that, in Figure S6, we show that the generated images are realistic and different from the ones found in the training set. For each generated image, we identify its nearest neighbor in the training data using the patch-level UNI embeddings [2], and show that they differ in shape and content. ZoomLDM can produce high-quality and unique samples for data-scarce magnifications, essentially avoiding memorization, by learning to synthesize images at all scales.

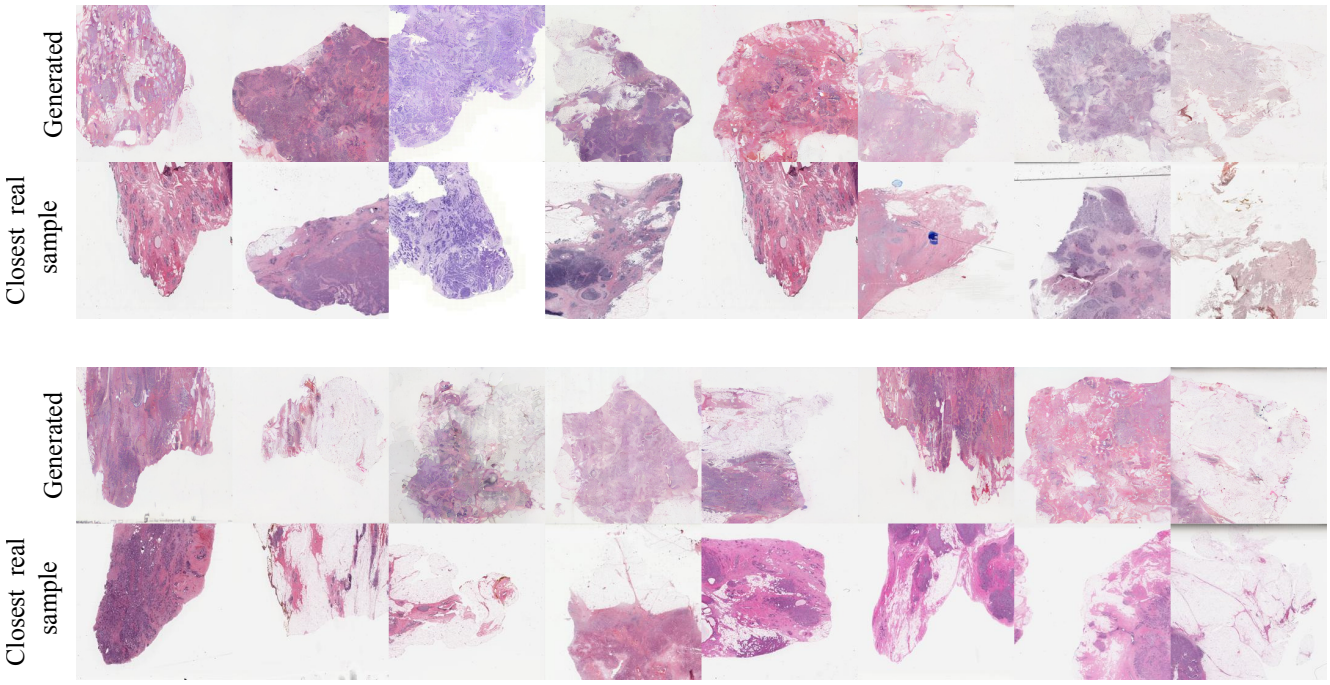

Figure S6. We present $0.15625\times$ images generated from our model and their nearest neighbors in the training dataset. Although only trained on $\sim$ 2500 images, our 400M parameter model did not memorize the training samples and successfully synthesized novel images at that magnification.

S4.3. Patches from all scales

In Figures S7 and S9, we showcase synthetic samples from ZoomLDM and the real images used to extract embeddings in histopathology and satellite. Samples from our model are realistic and preserve semantic features found in the reference patches. In data-scarce scenarios, such as $0.15625x$ magnification, achieving comparable image quality would be infeasible for a standalone model trained solely on that magnification (as indicated by the FIDs in Table 1 of the main text).

Interestingly, for magnifications below $5\times$ we find that the model can almost perfectly replicate the source image since the SSL embeddings used as conditioning contain enough information to reconstruct the patch at that scale perfectly. Although this may seem like a memorization issue, our experiments with the CDM in S4.2 show that our model has not just memorized the SSL embedding and image pairs. We believe that for these domains, this faithfulness to the conditions is advantageous as it can limit the hallucinations of the model, which are mostly unwanted in domains such as medical images.

S4.4. Large images

In Figures S10,S11 we present 4096×4096 px images generated from our histopathology and satellite ZoomLDM model. Readers can find more examples on histodiffusion.github.io/docs/projects/zoomldm.

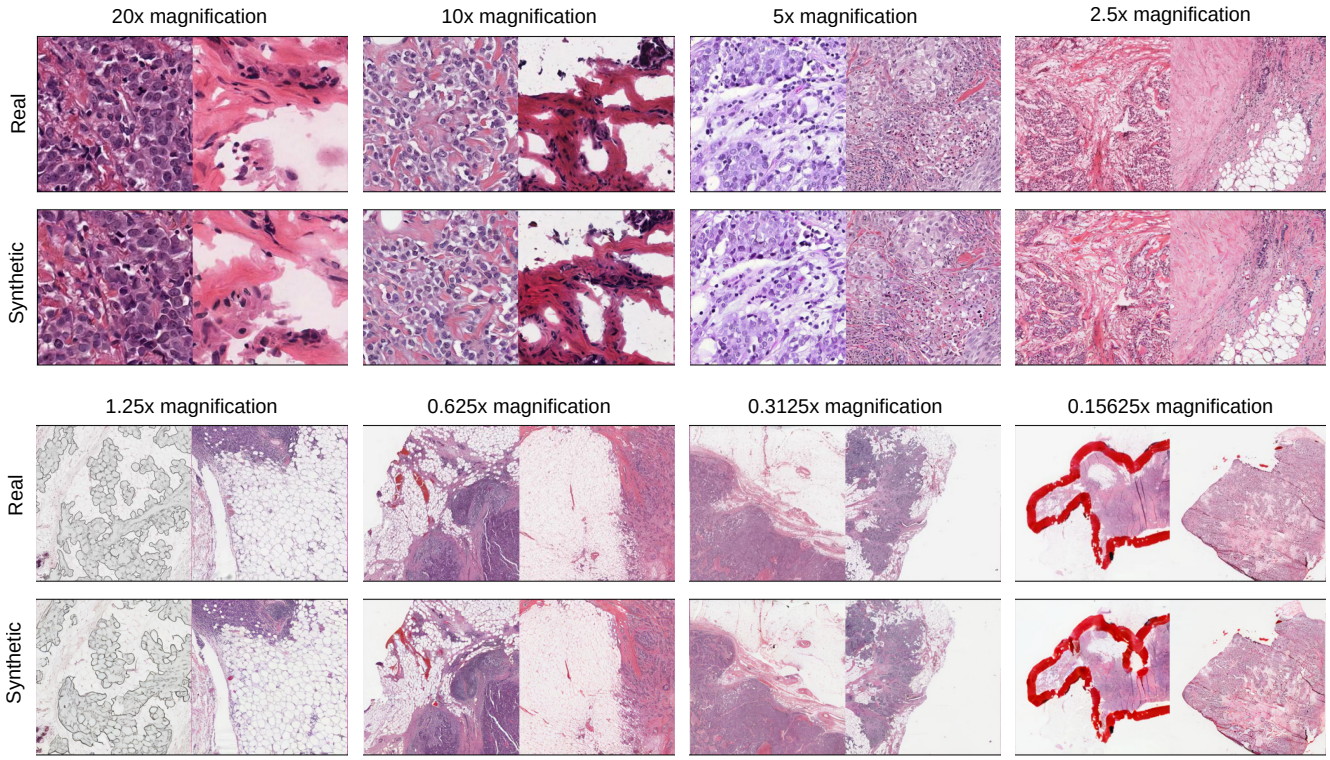

Figure S7. Synthetic patches (256×256 pixel) generated by ZoomLDM juxtaposed with the corresponding real images from TCGA-BRCA. Across all magnifications, ZoomLDM preserves the semantic features of the reference patches.

S4.5. Comparison to previous works

In Figure S12, we compare our method and previous works on a single example image. We extract SSL embeddings from the 4k to replicate this image as closely as possible. We highlight our two main differences with previous methods. The method of $\infty -$ Brush [9] retains some global structures but fails to produce any high-resolution details in the image. On the other hand, the patch-based model of [6] produces high-quality details but fails to capture large-scale structures that span more than a single patch. Our method solves both issues at the same time while maintaining a reasonable inference time, as discussed in the main text. We provide further comparisons to $\infty - Brush$ in Figure S13. Our generated images contain noticeably better detail.

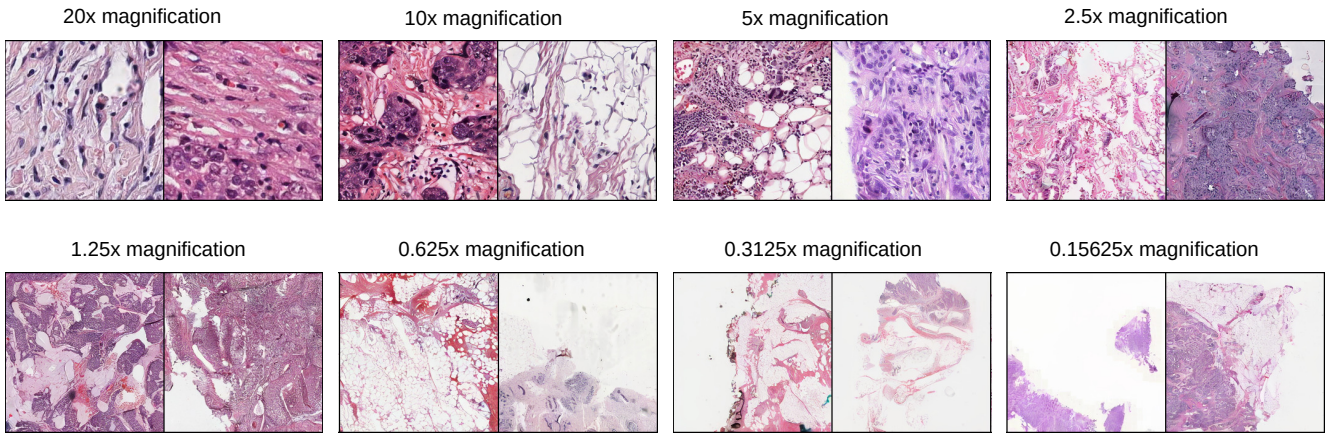

Figure S8. Images synthesized by ZoomLDM using conditions sampled from our Conditioning Diffusion model (CDM).

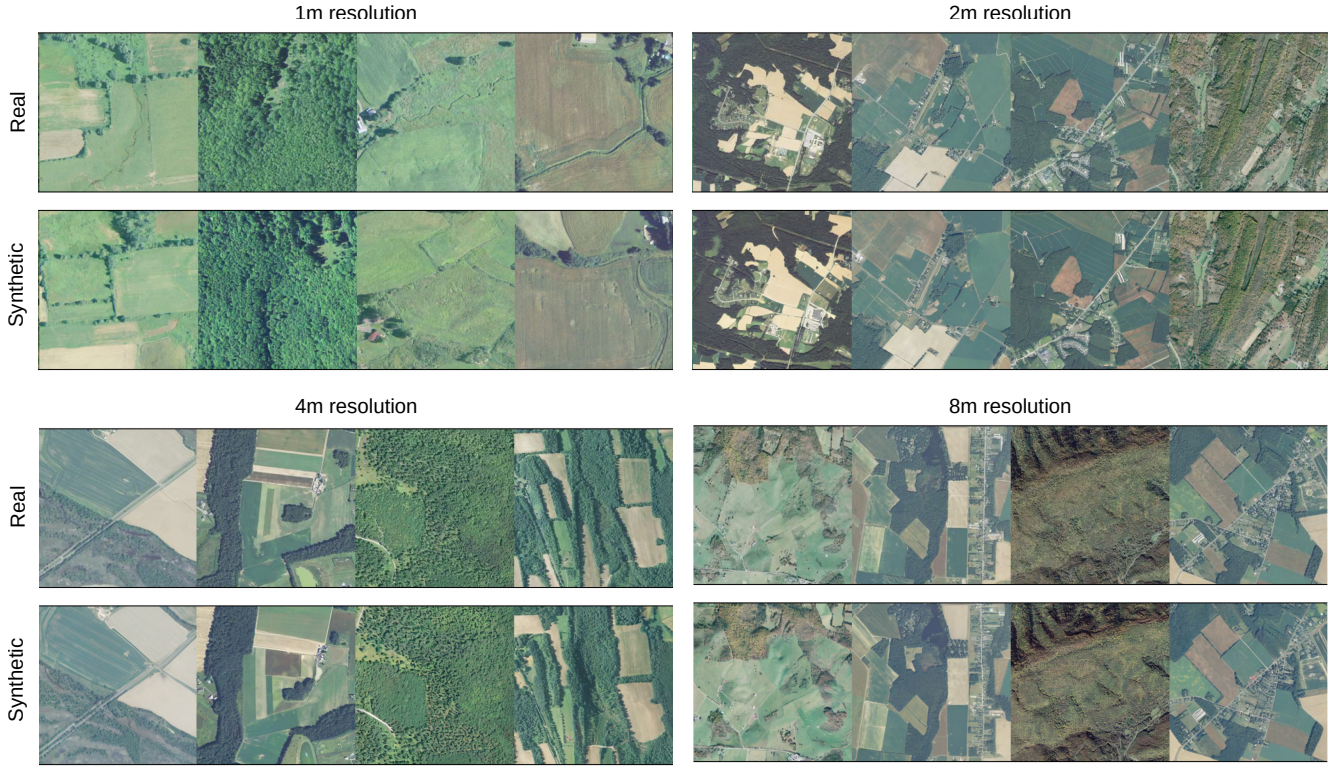

Figure S9. Synthetic patches (256×256 pixel) generated by ZoomLDM juxtaposed with the corresponding real images from NAIP

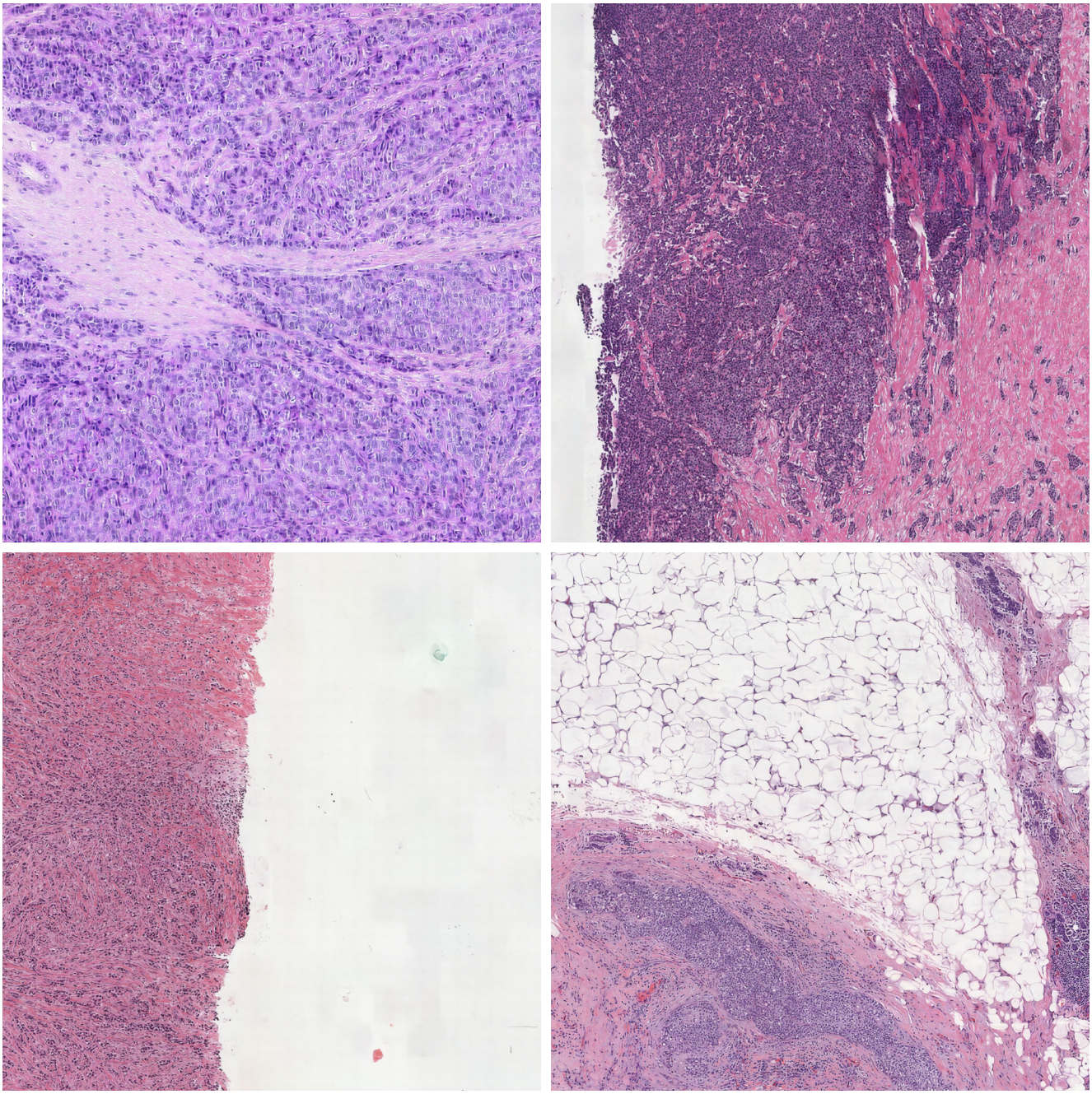

Figure S10. We present 4096×4096 images generated from our histopathology model. Our results exhibit correct global structures in terms of the arrangement of cells and tissue while also maintaining high-resolution details. We point out two weaknesses: The local model fails to maintain coherency for structures where the lower-scale image does not provide guidance, such as the thin structures in the bottom-right image. In addition, for large uniform areas, such as the background in the bottom left image, the 'stitching' of the generated $20\times$ patches is visible with noticeable discontinuities along their edges.

Figure S11. We present 4096×4096 images generated from our satellite model. The results demonstrate images with reasonable global structures that also maintain high-resolution features. A similar weakness to the pathology images is visible, with slight discontinuities among the high-resolution patch borders.

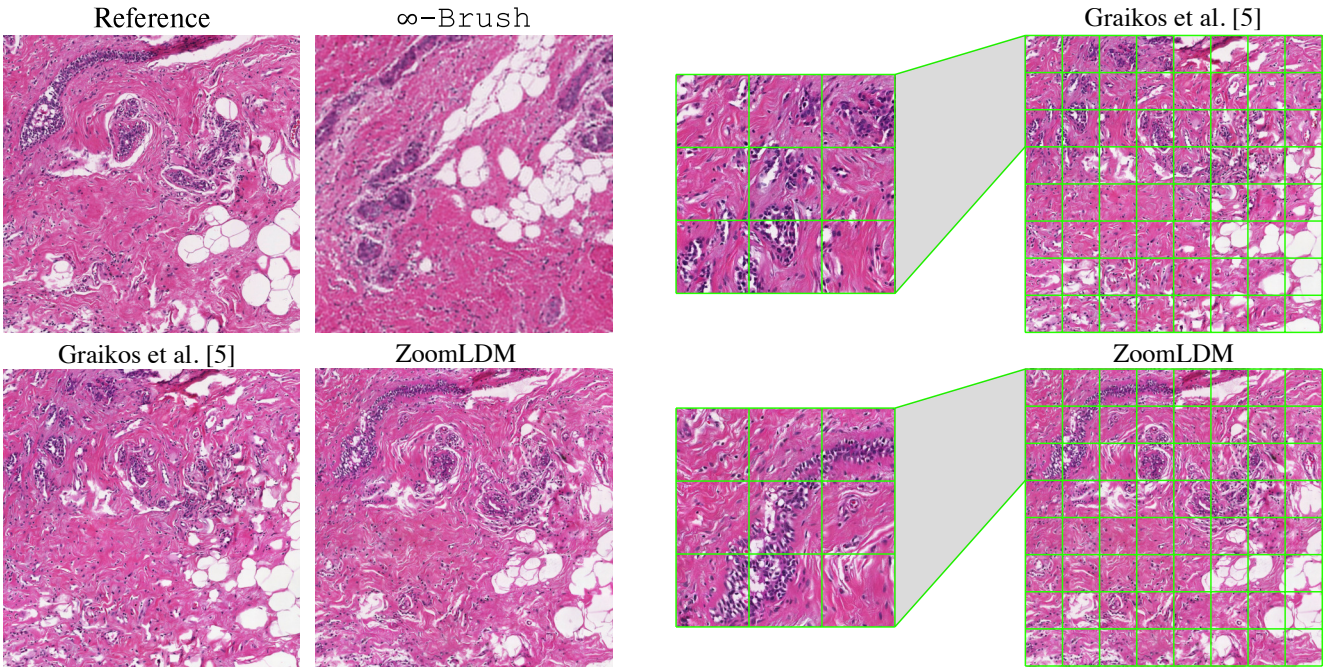

Figure S12. We compare with two recent previous methods that also generated large histopathology images. In this example, we compare a 2048×2048 image from $\infty -$ Brush and [6] to the same image generated from our model. We exceed both previous methods, with $\infty -$ Brush producing realistic global context but blurry details and [6] completely failing to capture larger scale structures.

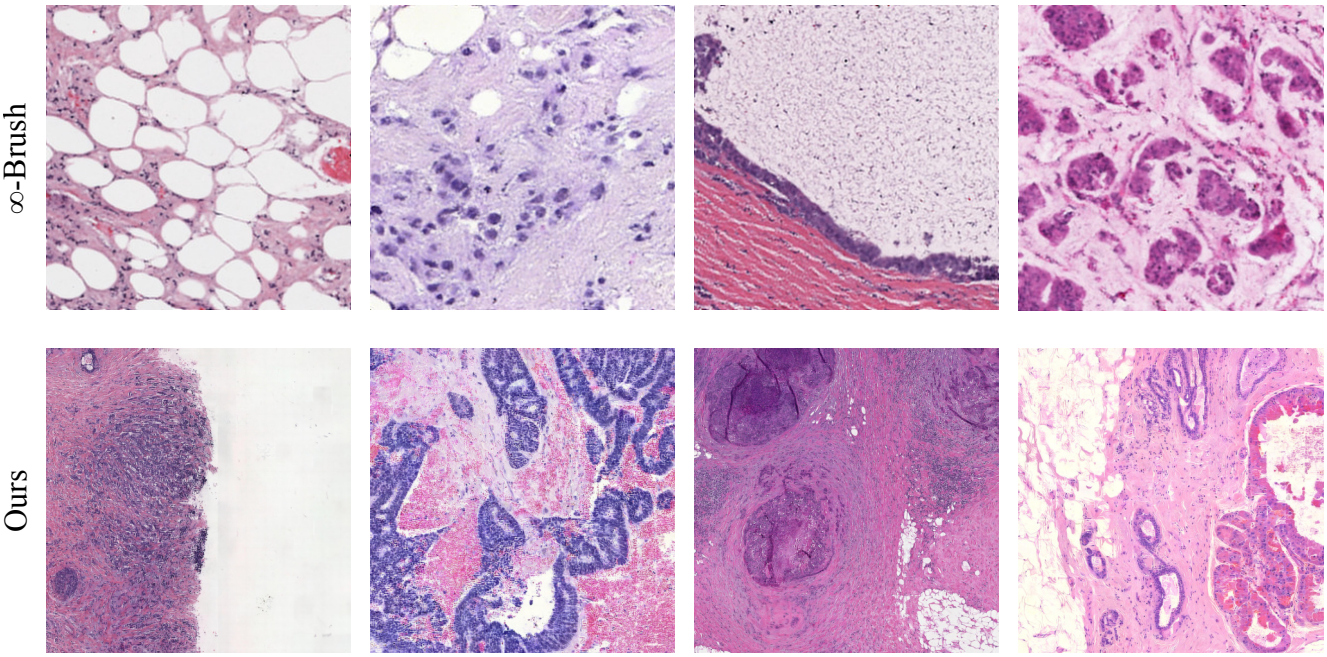

Figure S13. Comparison between $\infty -$ Brush [9] and our method.

References

[1] Richard J Chen, Chengkuan Chen, Yicong Li, Tiffany Y Chen, Andrew D Trister, Rahul G Krishnan, and Faisal Mahmood. Scaling vision transformers to gigapixel images via hierarchical self-supervised learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16144–16155, 2022. 1, 2

[2] Richard J Chen, Tong Ding, Ming Y Lu, Drew FK Williamson, Guillaume Jaume, Bowen Chen, Andrew Zhang, Daniel Shao, Andrew H Song, Muhammad Shaban, et al. Towards a general-purpose foundation model for computational pathology. *Nature Medicine*, 2024. 2, 7

[3] Hyungjin Chung, Jong Chul Ye, Peyman Milanfar, and Mauricio Delbracio. Prompt-tuning latent diffusion models for inverse problems. In *Forty-first International Conference on Machine Learning*, 2024. 4

[4] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit Haim Bermano, Gal Chechik, and Daniel Cohen-or. An image is worth one word: Personalizing text-to-image generation using textual inversion. In *The Eleventh International Conference on Learning Representations*, 2023. 4, 5

[5] Alexandros Graikos, Nebojsa Jojic, and Dimitris Samaras. Fast constrained sampling in pre-trained diffusion models. *arXiv preprint arXiv:2410.18804*, 2024. 3, 4

[6] Alexandros Graikos, Srikar Yellapragada, Minh-Quan Le, Saarthak Kapse, Prateek Prasanna, Joel Saltz, and Dimitris Samaras. Learned representation-guided diffusion models for large-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8532–8542, 2024. 1, 4, 8, 12

[7] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 4

[8] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 4

[9] Minh-Quan Le, Alexandros Graikos, Srikar Yellapragada, Rajarsi Gupta, Joel Saltz, and Dimitris Samaras. ∞-brush: Controllable large image synthesis with diffusion models in infinite dimensions, 2024. 1, 8, 12

[10] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 1

[11] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4195–4205, 2023. 2

[12] Caleb Robinson, Le Hou, Kolya Malkin, Rachel Soobitsky, Jacob Czawlytko, Bistra Dilkina, and Nebojsa Jojic. Large scale high-resolution land cover mapping with multi-resolution data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 12726–12735, 2019. 1

[13] USGS. National agriculture imagery program (NAIP), 2023. https://www.usgs.gov/centers/eros/science/usgs-eros-archive-aerial-photography-national-agriculture-imagery-program-naip. 1

[14] Jianyi Wang, Zongsheng Yue, Shangchen Zhou, Kelvin CK Chan, and Chen Change Loy. Exploiting diffusion prior for real-world image super-resolution. *International Journal of Computer Vision*, pages 1–21, 2024. 6, 7

[15] Zongsheng Yue, Jianyi Wang, and Chen Change Loy. Efficient diffusion model for image restoration by residual shifting. *arXiv preprint arXiv:2403.07319*, 2024. 6, 7

[16] Zongsheng Yue, Jianyi Wang, and Chen Change Loy. Resshift: Efficient diffusion model for image super-resolution by residual shifting. *Advances in Neural Information Processing Systems*, 36, 2024. 6, 7