Training-Free Inference for High-Resolution Sinogram Completion *

Jiaze E[1] and **Srutarshi Banerjee**[2] and **Tekin Bicer**[2,3] and **Guannan Wang**[1] and **Yanfu Zhang**[1], **Bin Ren**[1]
[1]William & Mary
[2]Argonne National Laboratory
[3]University of Chicago
je@wm.edu, {sruban, tbicer}@anl.gov, {gwang01. yzhang105, bren}@wm.edu

Abstract

High-resolution sinogram completion is critical for computed tomography reconstruction, as missing projections can introduce severe artifacts. While diffusion models provide strong generative priors for this task, their inference cost grows prohibitively with resolution. We propose HRSino, a training-free and efficient diffusion inference approach for high-resolution sinogram completion. By explicitly accounting for spatial heterogeneity in signal characteristics, such as spectral sparsity and local complexity, HRSino allocates inference effort adaptively across spatial regions and resolutions, rather than applying uniform high-resolution diffusion steps. This enables global consistency to be captured at coarse scales while refining local details only where necessary. Experimental results show that HRSino reduces peak memory usage by up to 30.81% and inference time by up to 17.58% compared to the state-of-the-art framework, and maintains completion accuracy across datasets and resolutions.

1 Introduction

A sinogram is a 2D projection-domain representation of computed tomography (CT) data, where each row corresponds to an acquisition angle and each column to a detector position. In practical CT reconstruction pipelines, projection data are often incomplete due to limited-angle acquisition, reduced sampling, or experimental constraints, and missing sinogram measurements can introduce severe artifacts that are amplified during reconstruction [Kalender, 2011]. Sinogram completion therefore plays a critical role in restoring projection data prior to image reconstruction.

High-resolution sinograms, often reaching 2048×2048 or beyond, arise in synchrotron and nano-scale CT systems, where dense angular sampling and fine detector resolution are required to preserve quantitative accuracy. In these settings, high resolution is required to preserve measurement fidelity and ensure stable reconstruction, and aggressive resolution reduction can fundamentally alter the inverse problem and compromise downstream analysis [Slaney and Kak, 1988].

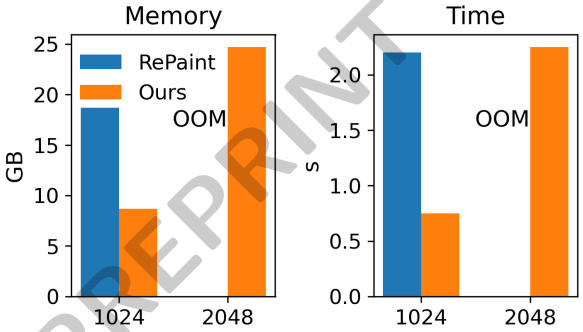

Figure 1: Peak GPU memory usage (left) and inference time (right) for sinogram completion at 1024×1024 and 2048×2048 resolutions. Unoptimized diffusion inference encounters out-of-memory (OOM) at high resolution, while our approach remains feasible and substantially reduces memory consumption and runtime.

Under the high-resolution setting described above, inference efficiency becomes a primary bottleneck for sinogram completion. Diffusion models have demonstrated strong performance in completion due to their ability to progressively refine structure through iterative denoising [Sohl-Dickstein *et al.*, 2015; Ho *et al.*, 2020; Saharia *et al.*, 2022; Lugmayr *et al.*, 2022]. However, this inference procedure requires repeatedly applying a large UNet over the full spatial domain across many denoising steps, leading to substantial memory and computational overhead as resolution increases. In practical high-resolution sinogram settings, this overhead translates into tens of gigabytes of GPU memory consumption and seconds of inference time, even when processing a single input at a time. Such costs quickly exceed the capacity of commonly available hardware, making high-resolution sinogram deployment challenging. As illustrated in Fig 1, unoptimized diffusion inference exhibits prohibitive memory consumption and runtime at 2048×2048 resolution, resulting in out-of-memory (OOM) failures.

A number of approaches have been proposed to improve diffusion efficiency. Step reduction and sampling acceleration methods aim to shorten the denoising trajectory by modifying the global sampling schedule [Lu *et al.*, 2022; Sali-

*Supplementary material and an extended version of this paper are available at: https://arxiv.org/abs/2506.08809

mans and Ho, 2022; Meng *et al.*, 2023]. Other efforts focus on architectural or implementation-level optimizations to reduce per-step computation or memory usage [Li *et al.*, 2023; Zhang *et al.*, 2024b; Zhu *et al.*, 2024]. While effective in reducing overall inference cost, these methods typically apply computation uniformly across the entire spatial domain and maintain a fixed denoising depth for all regions. As a result, computational effort is often wasted on regions that require little refinement, leaving substantial redundancy in high-resolution diffusion inference under strict memory constraints. This limitation becomes particularly pronounced in high-resolution sinogram completion, where large spatial regions exhibit low signal variation and do not require uniform refinement across all diffusion steps.

Motivated by this perspective, we propose HRSino, a training-free efficient high-resolution sinogram completion framework that reorganizes diffusion inference along both spatial and temporal dimensions by selectively allocating computation across regions and adapting denoising depth to local complexity. Rather than modifying model architecture or retraining diffusion networks, HRSino operates entirely at inference time by reallocating computation to where it is most needed. To enable patch-wise diffusion inference without full-frame activation, we employ a resolution-guided inference scheme that extracts coarse global structure at a reduced resolution and uses it to guide localized refinement at higher resolutions. By avoiding denoising on the full-resolution grid from the outset, this strategy substantially reduces peak memory usage while preserving global consistency. Unlike prior approaches that introduce global embeddings, conditioning tokens, or architectural modifications to maintain context [Avrahami *et al.*, 2022; Zhang *et al.*, 2023a; Tumanyan *et al.*, 2023; Zhang *et al.*, 2023b], this mechanism operates entirely at inference time and requires no retraining.

In practice, a significant portion of redundancy in high-resolution diffusion inference comes from large variations in local signal complexity across the spatial domain. Large portions of the input often exhibit smooth variation and limited high-frequency content [Slaney and Kak, 1988], yet are subjected to the same inference cost as structurally rich regions. To address this imbalance, HRSino introduces a frequency-aware patch-skipping mechanism that identifies low-energy patches using a simple FFT-based score. Patches with negligible spectral content are bypassed during diffusion inference and approximated using compact representations, thereby eliminating redundant computation without compromising completion fidelity.

For the remaining patches, structural complexity can vary substantially, ranging from flat gradients to fine edges and high-frequency details. Applying a fixed number of denoising steps to all patches therefore leads to either over-computation in simple regions or insufficient refinement in complex ones. To better align inference cost with local content richness, HRSino employs a structure-adaptive denoising scheduler that assigns each patch a customized number of denoising steps. This allocation is determined by a joint complexity score combining Shannon entropy [Shannon, 1948] and frequency energy, capturing both spatial variability and spectral content. Together, frequency-aware patch skipping and adaptive step allocation enable diffusion inference to concentrate computational effort on structurally informative regions, while substantially reducing redundant computation under strict constraints.

Together, these designs preserve completion fidelity by first capturing global structure that maintains long-range consistency across resolution levels. Patch skipping then removes only regions with negligible signal, ensuring that no structurally relevant information is lost, while adaptive denoising assigns each patch a sufficient number of steps based on its complexity, ensuring adequate refinement in detail-rich areas. As a result, HRSino improves efficiency (both in peak memory usage and inference speed) without compromising completion quality. Overall, our contributions are as follows:

- We present HRSino, a novel inference-time diffusion framework for high-resolution sinogram completion under strict constraints. The framework reorganizes diffusion inference to enable efficient high-resolution completion without retraining or architectural modifications.
- We introduce a pair of inference-time mechanisms: frequency-aware patch skipping and structure-adaptive denoising, which enable spatially adaptive allocation of diffusion computation. By selectively skipping low-information regions and assigning different denoising depths based on local complexity, these mechanisms address the redundancy inherent in uniform full-resolution diffusion inference.
- HRSino enables 2048×2048 sinogram completion on an A100 GPU, reducing peak memory usage by up to 30.81% and inference time by up to 17.58% without degrading completion quality across multiple datasets, input resolutions, mask types, and mask ratios.

2 Related Work

Deep Learning Based Completion Methods. Early approaches employ CNN-, GAN-, or transformer-based architectures. Representative examples include ICT [Wan *et al.*, 2021] which leverages transformer attention to model long-range structures and produces pluralistic high-fidelity completion results. Recently, diffusion-based models have achieved strong performance on image completion due to their generative capacity and ability to model uncertainty. Methods such as RePaint [Lugmayr *et al.*, 2022], Palette [Saharia *et al.*, 2022], Blended Diffusion [Avrahami *et al.*, 2022], and CoPaint [Zhang *et al.*, 2023a] enhance global coherence on images through iterative sampling or semantic priors. Sinogram completion has also been studied in industrial and medical CT settings using CNNs [Lee *et al.*, 2018; Jin *et al.*, 2017] and transformer-based such as SinoTx [E *et al.*, 2025] for sparse-view recovery or artifact reduction, but these works generally ignore memory constraints and inference time optimizations. End-to-end CT reconstruction methods [Guo *et al.*, 2025] directly map incomplete sinograms to reconstructed images. While effective for reducing image-space artifacts, they do not provide sinogram-domain control or modular compatibility with existing CT reconstruction workflows.

Inference Acceleration in Diffusion Models. Improving the efficiency of diffusion inference has become a growing area of focus. Sampling-based methods reduce denoising steps through distillation [Salimans and Ho, 2022], or step-aware training [Xiao *et al.*, 2024]. Architectural methods employ memory-aware attention, such as in HiDiffusion [Zhang *et al.*, 2024a] and DiffIR [Xia *et al.*, 2023]. Model pruning and lightweight design strategies have also been proposed, as seen in SnapFusion [Li *et al.*, 2023], EcoDiff [Zhang *et al.*, 2024b], and DiP-GO [Zhu *et al.*, 2024]. Patch-wise generation and conditional masking [Avrahami *et al.*, 2023] have also been explored to control computation scope. However, most existing methods require retraining, loss rebalancing, or internal modifications to the model. Many of these approaches are orthogonal to our work and can be integrated in principle, but they do not account for and apply to the structural characteristics, such as directional sparsity and local interpretability. Applying them effectively would require additional adaptation.

Memory and Runtime Efficient Deep Learning. Beyond diffusion, memory- and latency-aware inference strategies have been widely studied in deep learning. Techniques include recomputation and memory reuse [Chen *et al.*, 2016; Jain *et al.*, 2020], and runtime-adaptive depth scaling [Xia *et al.*, 2021]. MEST [Yuan *et al.*, 2021] also explore sparsity-aware training or deployment under tight memory budgets. These techniques address memory bottlenecks in standard feedforward networks and our work focuses instead on iterative generative models with different constraints.

3 HRSino: High-Resolution Sinogram Completion

As shown in Fig 2, HRSino builds upon RePaint [Lugmayr *et al.*, 2022] as its default backbone and restructures the inference process into a three-stage resolution-guided pipeline. The sinogram is first denoised at low resolution to establish global structure, then refined at mid resolution, and finally completed at full resolution through patch-wise inference. At each stage, the upsampled output from the previous resolution is fused with the current input before denoising, ensuring hierarchical guidance across scales. This hierarchical design avoids full-frame activation and substantially reduces memory usage while preserving long-range consistency. To further improve efficiency under the structural characteristics of sinograms, HRSino introduces two inference-time modules: (1) frequency-aware patch skipping, which exploits the spectral sparsity of background regions to bypass redundant computation, and (2) structure-adaptive step allocation, which leverages local structural heterogeneity to adjust denoising depth per patch.

3.1 Resolution-Guided Progressive Inference

To efficiently complete high-resolution sinograms while avoiding memory overflow, HRSino adopts a three-stage progressive inference pipeline operating over low, mid, and high resolutions. Let x_r denote the input at resolution $r \in \{low, mid, high\}$. Each resolution level x_r is obtained by downsampling the original sinogram using a fixed ratio: low and mid-resolution inputs correspond to $0.25\times$ and $0.5\times$ the original resolution, respectively. At the first stage, full DDIM inference is performed on x_{low} to generate a coarse prior $\hat{x}_{low}$ to capture global structural cues at minimal memory cost.

In the second stage, we refine the geometry at mid-resolution. x_{mid} is fused with the upsampled output from the previous stage using a resolution-aware weighted sum:

$$\tilde{x}_{mid} = \frac{1}{2} \cdot (Up(\hat{x}_{low}) + x_{mid}), \tag{1}$$

where $Up(\cdot)$ denotes nearest-neighbor upsampling. This fusion helps the model retain mid-level structure while reinforcing global context. The resulting $\tilde{x}_{mid}$ is then denoised via DDIM to produce $\hat{x}_{mid}$.

At the final stage, we operate on the original full-resolution input. We divide x_{high} into patches $\{x^i_{high}\}$. Each patch is of fixed size, ensuring consistent inference cost per patch across resolutions. For each patch, we retrieve the aligned region from $\hat{x}_{mid}$, upsample it, and fuse it with the local patch:

$$\tilde{x}^i_{high} = \frac{1}{2} \cdot (Up(\hat{x}^i_{mid}) + \cdot x^i_{high}). \tag{2}$$

Each $\tilde{x}^i_{high}$ is then denoised via DDIM. Sec 3.2 and Sec 3.3 introduce more details about the final stage. This patch-wise processing avoids full-frame memory load while preserving long-range consistency through hierarchical conditioning. Each patch is processed sequentially during inference to ensure bounded memory usage.

3.2 Frequency-Aware Patch Skipping

High-resolution sinograms often contain broad low-frequency backgrounds with narrow high-frequency details. To reduce computation in low-complexity regions, HRSino introduces frequency-aware patch skipping based on localized spectral content.

Frequency-Aware Patch Pruning

To reduce unnecessary computation in low-information regions of high-resolution sinograms, we introduce a mask-aware frequency-based patch skipping strategy. Let P denote a high-resolution sinogram patch, and let $\mathcal{F}(P)$ represent its real-valued 2D Fourier transform. We first compute the high-frequency energy ratio $\gamma(P)$ as:

$$\gamma(P) = \frac{\sum_{(u,v)\in\Omega_{high}} |\mathcal{F}(P)_{u,v}|^2}{\sum_{(u,v)} |\mathcal{F}(P)_{u,v}|^2}, \tag{3}$$

where Ω_{high} is a predefined high-pass band, typically the outer third of the frequency spectrum. In addition, we measure the mask ratio r. To incorporate mask awareness, we define the adjusted score as:

$$\gamma'(P) = (1 - r(P)) \cdot \gamma(P) + \tau \cdot r(P), \tag{4}$$

where τ is a small constant that safeguards against skipping heavily masked patches. A patch is considered redundant and skipped if its adjusted score $\gamma'(P)$ falls below a threshold τ. In this way, structurally smooth or empty regions can still be skipped, while patches with large missing areas are preserved for reliable completion.

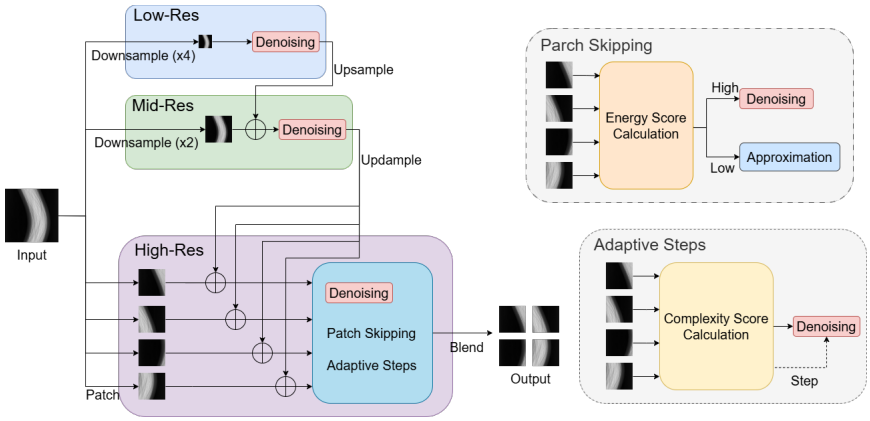

Figure 2: Overview of HRSino. The left illustrates the three-stage resolution-guided pipeline: low resolution followed by mid and high resolutions in a progressive refinement scheme, with the final stage performed patch-wise for detail recovery. At each stage, the upsampled output from the previous resolution is fused with the current input before denoising. The right zooms into the high-resolution stage, where two modules are applied: frequency-aware patch skipping (bypassing low-information patches) and structure-adaptive step allocation (assigning variable denoising steps by patch complexity).

Rather than running full DDIM inference for spectrally sparse patches, we replace their outputs with a fixed approximation. It is obtained by passing a synthetic input patch—filled with low-amplitude Gaussian noise with mean 0, and standard deviation 0.01—through the original RePaint and DDIM steps. This design simulates typical background regions. It avoids repeated computation in trivial regions and ensures consistency with standard outputs.

Cosine-Based Patch Blending

To avoid visual artifacts at patch boundaries, particularly when adjacent regions are inferred with different mechanisms, we apply a smooth blending operation between neighboring patches. This is especially important in high-resolution sinograms where directional continuity must be preserved.

Let P_1 and P_2 denote two horizontally adjacent reconstructed patches, and $p \in [0, L]$ be local coordinate across boundary region of width L pixels. Define a cosine-based spatial weight function:

$$\alpha(p) = \frac{1}{2}\left(1 - cos(\frac{\pi p}{L})\right), \tag{5}$$

which increases smoothly from 0 to 1 across the boundary. The blended pixel value is computed as:

$$P_{blend}(x, y) = \alpha(p) \cdot P_1(x, y) + (1 - \alpha(p)) \cdot P_2(x, y). \tag{6}$$

This produces a smooth interpolation where the transition is soft and visually imperceptible, reducing abrupt intensity jumps at patch edges.

We apply this blending only when necessary. Specifically, we compute the average gradient magnitude in the overlapping region using a standard Sobel filter applied to the corresponding mid-resolution sinogram. If the gradient exceeds a predefined threshold η, blending is enabled; otherwise, a simple hard stitch is used to save computation in flat areas. This conditional mechanism ensures efficiency without compromising visual consistency in structurally complex regions.

3.3 Structure-Adaptive Denoising

To further reduce computation, we allocate a different number of denoising steps to each patch based on its structural complexity. Unlike static diffusion sampling where every region receives the same number of DDIM steps, we adapt the inference depth dynamically to match the content difficulty.

Complexity Score Computation

Let P denote a patch in the high-resolution sinogram. We define a complexity score $\kappa(P)$ that combines two frequency-domain measures: Shannon entropy and high-frequency energy. The Shannon entropy $\mathcal{H}(P)$ is computed over the pixel intensity histogram:

$$\mathcal{H}(P_i) = -\sum_j p_j \log p_j, \tag{7}$$

where p_j is the normalized count of pixel intensity in bin i. This captures texture randomness and distribution uniformity.

Also, we compute the spectral energy using the L1-norm of the 2D FFT of the patch. The overall complexity score is defined as:

$$\kappa_i = \mathcal{H}(P_i) + \log(1 + ||\mathcal{F}(P_i)||_1), \tag{8}$$

where $\mathcal{F}(P_i)$ is the 2D real-valued FFT of the patch and $||\cdot||_1$ denotes the L1-norm of the spectrum.

Entropy captures randomness and texture variation in the spatial domain, while the FFT energy captures signal richness and directional complexity. Their combination provides a robust, domain-agnostic proxy for structural difficulty.

Patch-Wise Step Mapping

To convert κ_i into a patch-specific number of denoising steps, we apply a sigmoid-based scaling function:

$$S_i = \lfloor S_{min} + (S_{max} - S_{min}) \cdot \sigma(\kappa_i - \mu)) \rfloor \tag{9}$$

where μ is the mean complexity across all patches in the sinogram; $\sigma(\cdot)$ is the standard sigmoid function; S_{min}, S_{max} define the step range; $\lfloor \cdot \rfloor$ denotes rounding down to the nearest integer.

Method	2048×2048		1024×1024	
	Mem	Inf time	Mem	Inf time
RePaint	OOM		18.7	2.20
MCG	OOM		23.2	2.81
DiffIR	OOM		22.0	2.54
TD-Paint	OOM		19.3	1.48
HiDiffusion	35.7	2.73	12.5	0.92
HRSino (ours)	**24.7**	**2.25**	**8.7**	**0.75**
SinoTx	OOM		19.5	0.38
ICT	OOM		17.0	0.20
Spline (CPU)	0.4	0.22	0.4	0.07
Bilinear (CPU)	0.3	0.12	0.3	0.04

Table 1: Peak GPU memory (GB) and inference time (s) of different methods on `TomoBank` with random masks (ratio = 0.8) at resolutions 2048×2048 and 1024×1024. "OOM" indicates out-of-memory. Results on `LoDoPaB` exhibit nearly identical patterns in both memory and runtime and are omitted here for brevity.

This ensures a soft but data-driven allocation: patches with below-average complexity receive fewer sampling steps, while others are modeled more deeply. Using the sigmoid ensures that step variation remains smooth, differentiable, and robust to outliers. The centering around the mean further normalizes the step distribution across sinograms with varying overall complexity.

4 Evaluation

The evaluation has two main objectives: (1) demonstrating that HRSino significantly improves memory efficiency and inference speed while maintaining completion quality (Sec 4.2); (2) conducting ablation studies to validate the effectiveness of our mechanisms, including multi-resolution progressive inference, frequency-aware patch skipping and structure-adaptive denoising (Sec 4.3).

4.1 Experimental Setup

Evaluation Platform and Settings. All experiments are performed on a single NVIDIA A100 GPU with 40 GB on-device memory from the Polaris supercomputer of Argonne Leadership Computing Facility (ALCF) resources at Argonne National Laboratory (ANL), using CUDA 12.2, and PyTorch 2.1.0. All implementations employ PyTorch built-in optimizations: `torch.compile()` to enable graph-mode execution, mixed-precision inference via `torch.autocast()` (fp16), and `cudnn.benchmark = True` for kernel tuning.

All diffusion-based experiments use the denoising diffusion implicit models (DDIM) [Song *et al.*, 2020] sampler with 50 steps. This step count is consistent with values evaluated in DDIM paper, where 50-step sampling achieves a strong balance between quality and efficiency. All inferences are performed with a batch size of 1. For high-resolution inference, we partition sinograms into overlapping patches with patch size $P = 128$, overlap $O = 32$, and stride $S = 96$. This setting balances efficiency and contextual coverage, while overlap mitigates boundary artifacts. For patch skipping, the high-frequency threshold is set to $\tau = 0.08$, which empirically filters low-information patches without affecting completion fidelity. For adaptive step scheduling, we use $(S_{min}, S_{max}) = (10, 50)$ to stay within the DDIM step budget while allowing fine-grained inference depth. More details about hyperparameters are in the extended version.

To ensure fair comparison, all baseline models are retrained from scratch using their official codebases. Since HRSino operates entirely at inference time, no retraining or task-specific fine-tuning is applied. Each dataset provides 100k samples, with 80% used for training at a resolution of 512×512, 10% for validation, and 10% for testing.

Dataset. We evaluate two real-world datasets `TomoBank` and `LoDoPaB`. `TomoBank` [De Carlo *et al.*, 2018] consists of real sinogram images obtained from various materials and objects, derived from actual synchrotron radiation CT experiments, including Advanced Photon Source (APS) at ANL. It also includes samples with dynamic features [Mohan *et al.*, 2015] and in situ [Pelt and Batenburg, 2013] measurements, which capture a wide range of experiments at APS and other synchrotron facilities. To ensure consistency, we follow the official TomoPy [Gürsoy *et al.*, 2014] preprocessing pipeline, which includes ring artifact removal, rotation center alignment, and normalization of intensities to the [0,1] range. Projection counts, angular sampling, and rotation centers are configured according to the metadata provided in TomoBank. `LoDoPaB` [Leuschner *et al.*, 2021] is derived from the publicly available LIDC-IDRI lung CT database [Armato III *et al.*, 2011]. It provides simulated sinograms generated with a fixed parallel-beam geometry (1,000 projection angles, 513 detector bins), paired with real patient CT images. Unlike `TomoBank`, where sinograms are physically measured, `LoDoPaB` contains numerically generated sinograms with official splits provided via the DIVAL benchmark library [Leuschner and others, 2025]. To simulate realistic sparse-view CT scenarios, we apply two types of angular masks: (1) random masks, which discard a random subset of projection angles, and (2) periodic masks, which regularly subsample angles at fixed intervals. The mask ratio denotes the fraction of angles removed (e.g., 0.8 means only 20% of the projection angles are retained). In this work, we mainly evaluate challenging cases with mask ratios of 0.6 and 0.8, where the majority of angular information is missing.

Metrics. We evaluate each method using both efficiency and quality metrics. For computational performance, we report peak GPU memory usage and inference runtime, measured over multiple forward passes. Peak memory reflects the maximum GPU allocation during inference, indicating worst-case hardware demand. For completion fidelity, we use Structural Similarity Index (SSIM) [Wang *et al.*, 2004] and Peak Signal-to-Noise Ratio (PSNR). SSIM captures perceptual and structural consistency, while PSNR quantifies absolute pixel-wise differences. All images are normalized to the [0, 1] range before metric computation. SSIM is computed in single-scale mode using a 11×11 Gaussian kernel ($\sigma = 1.5$). All metrics are computed on grayscale images.

Baselines. We evaluate HRSino against representative baselines, including diffusion-based models optimized for high-resolution completion, transformer-based methods, and classical techniques. Among diffusion models, we consider Re-

Method	Mask	TomoBank		LoDoPaB	
		Ratio = 0.6	Ratio = 0.8	Ratio = 0.6	Ratio = 0.8
		2048×2048			
HRSino	Random	**0.930** (0.916) / **30.9** (29.9)	**0.927** (0.913) / **30.6** (29.7)	**0.940** (0.926) / **31.6** (30.4)	**0.935** (0.924) / **31.3** (30.4)
	Periodic	**0.926** (0.913) / **30.6** (29.7)	**0.924** (0.910) / **30.3** (29.4)	**0.937** (0.922) / **31.3** (30.4)	**0.934** (0.924) / **31.0** (30.1)
HiDiffusion	Random	0.903 (0.889) / 29.1 (28.2)	0.900 (0.886) / 28.8 (27.9)	0.911 (0.899) / 29.8 (28.9)	0.910 (0.893) / 29.5 (28.6)
	Periodic	0.902 (0.887) / 28.9 (28.0)	0.897 (0.883) / 28.6 (27.7)	0.912 (0.898) / 29.6 (28.7)	0.904 (0.892) / 29.1 (28.3)
Spline	Random	0.702 (0.688) / 23.8 (22.9)	0.698 (0.684) / 23.5 (22.6)	0.714 (0.694) / 24.5 (23.6)	0.711 (0.690) / 24.3 (23.4)
	Periodic	0.698 (0.681) / 23.7 (22.6)	0.694 (0.680) / 23.2 (22.3)	0.709 (0.694) / 24.4 (23.5)	0.703 (0.690) / 23.9 (23.1)
Bilinear	Random	0.695 (0.681) / 22.5 (21.7)	0.691 (0.677) / 22.3 (21.4)	0.703 (0.695) / 23.3 (22.6)	0.703 (0.687) / 22.9 (22.2)
	Periodic	0.695 (0.677) / 22.1 (21.4)	0.687 (0.673) / 21.9 (21.1)	0.703 (0.686) / 22.8 (22.0)	0.695 (0.681) / 22.3 (21.5)
		1024×1024			
HRSino	Random	**0.932** (0.919) / 31.2 (30.4)	0.927 (0.911) / **30.9** (30.0)	**0.944** (0.929) / **32.1** (31.3)	**0.936** (0.924) / 31.1 (30.5)
	Periodic	**0.929** (0.916) / 30.9 (30.1)	0.924 (0.913) / 30.6 (29.8)	0.937 (0.924) / 31.5 (30.8)	0.935 (0.922) / 30.9 (30.2)
RePaint	Random	**0.932** (0.919) / **31.4** (30.6)	**0.928** (0.915) / **30.9** (30.2)	**0.944** (0.927) / **32.1** (31.4)	**0.936** (0.925) / **31.2** (30.4)
	Periodic	**0.929** (0.914) / **31.1** (30.3)	**0.925** (0.912) / **30.8** (30.0)	**0.938** (0.924) / **31.8** (31.1)	**0.934** (0.924) / **31.0** (30.5)
HiDiffusion	Random	0.902 (0.888) / 29.8 (29.0)	0.898 (0.884) / 29.4 (28.6)	0.914 (0.900) / 30.7 (29.8)	0.909 (0.896) / 30.3 (29.5)
	Periodic	0.899 (0.885) / 29.5 (28.7)	0.895 (0.881) / 29.1 (28.3)	0.910 (0.896) / 30.4 (29.6)	0.906 (0.892) / 29.9 (29.1)
MCG	Random	0.901 (0.887) / 29.9 (29.1)	0.896 (0.883) / 29.4 (28.7)	0.914 (0.899) / 30.6 (29.9)	0.908 (0.894) / 30.2 (29.5)
	Periodic	0.897 (0.884) / 29.5 (28.8)	0.894 (0.880) / 29.1 (28.3)	0.909 (0.895) / 30.1 (29.4)	0.904 (0.891) / 29.8 (29.0)
DiffIR	Random	0.893 (0.881) / 29.4 (28.6)	0.889 (0.877) / 29.0 (28.2)	0.905 (0.892) / 30.2 (29.4)	0.900 (0.888) / 29.8 (29.0)
	Periodic	0.890 (0.878) / 29.1 (28.3)	0.886 (0.874) / 28.7 (27.9)	0.901 (0.889) / 29.7 (28.9)	0.897 (0.885) / 29.3 (28.5)
SinoTx	Random	0.880 (0.866) / 28.8 (28.0)	0.876 (0.862) / 28.4 (27.6)	0.891 (0.877) / 29.6 (28.7)	0.887 (0.873) / 29.1 (28.2)
	Periodic	0.877 (0.863) / 28.5 (27.7)	0.873 (0.859) / 28.1 (27.3)	0.888 (0.873) / 29.2 (28.4)	0.884 (0.870) / 28.7 (27.9)
ICT	Random	0.872 (0.858) / 28.5 (27.6)	0.868 (0.854) / 28.0 (27.2)	0.882 (0.869) / 29.0 (28.2)	0.876 (0.864) / 28.6 (27.7)
	Periodic	0.869 (0.855) / 28.1 (27.3)	0.865 (0.851) / 27.7 (26.9)	0.880 (0.866) / 28.7 (27.9)	0.873 (0.862) / 28.3 (27.4)
Spline	Random	0.718 (0.704) / 24.7 (23.8)	0.713 (0.699) / 24.3 (23.4)	0.728 (0.714) / 25.3 (24.4)	0.723 (0.710) / 24.9 (24.2)
	Periodic	0.715 (0.701) / 24.4 (23.5)	0.710 (0.696) / 24.0 (23.1)	0.726 (0.711) / 25.0 (24.1)	0.721 (0.707) / 24.5 (23.7)
Bilinear	Random	0.711 (0.697) / 23.4 (22.6)	0.707 (0.693) / 23.0 (22.2)	0.722 (0.708) / 24.1 (23.2)	0.718 (0.702) / 23.7 (22.7)
	Periodic	0.708 (0.694) / 23.1 (22.3)	0.704 (0.690) / 22.7 (21.9)	0.718 (0.704) / 23.7 (22.8)	0.714 (0.700) / 23.3 (22.4)

Table 2: SSIM / PSNR on `TomoBank` and `LoDoPaB` datasets under random and periodic masks (ratio = 0.6, 0.8) at resolutions 2048×2048 and 1024×1024. Each entry is formatted as: completed sinogram (CT reconstruction by FBP).

Paint [Lugmayr *et al.*, 2022], a general-purpose completion model that serves as the backbone for HRSino; HiDiffusion [Zhang *et al.*, 2024a], which improves memory efficiency through resolution-aware U-Net design and attention; DiffIR [Xia *et al.*, 2023], which introduces a intermediate prior to reduce sampling steps for image restoration; MCG [Chung *et al.*, 2022], which improves diffusion sampling stability via gradient-based manifold consistency guidance; and TD-Paint [Mayet *et al.*, 2025], which accelerates diffusion-based completion by reusing known regions across diffusion timesteps without architectural modification. For transformer-based approaches, we include SinoTx [E *et al.*, 2025], tailored to sinogram completion, and ICT [Wan *et al.*, 2021], designed for efficient completion. Finally, Bilinear and Spline interpolation serve as classical non-learning baselines.

4.2 Overall Quantitative and Qualitative Results

Peak Memory Usage Comparison

As shown in Tab 1, HRSino significantly reduces peak GPU memory usage compared to prior methods across both tested resolutions. RePaint, DiffIR, MCG, and TD-Paint all encounter out-of-memory (OOM) failures at 2048×2048 resolution, indicating that standard diffusion pipelines cannot scale to such resolutions even with a high-end GPU. In contrast, HRSino remains fully operational under these conditions and achieves up to a 30.81% reduction in memory usage relative to the most efficient baseline HiDiffusion. Notably, this trend holds consistently across input sizes, indicating that our framework scales gracefully.

Inference Time Comparison

As shown in Tab 1, HRSino consistently achieves faster inference than the fastest diffusion-based baseline, HiDiffusion, with speed up to 17.58% at 2048×2048 resolution. These improvements stem from two sources: progressive scheduling reduces redundant steps in simple regions, while patch skipping eliminates computation entirely for spectrally sparse areas. We further validate the effectiveness of these two mechanisms in the ablation studies (Sec 4.3). This demonstrates that high-resolution completion can be accelerated without compromising completion quality.

Completion Quality

Tab 2 summarizes accuracy results. At 2048×2048 resolution, HRSino achieves the best performance among all baselines while remaining memory-efficient, demonstrating its ability to extend high-quality completion to resolutions where other diffusion models fail. At 1024×1024, HRSino delivers accuracy comparable to its computation-intensive counterpart RePaint, showing that our optimizations do not compromise fidelity at moderate scales. Compared to DiffIR, MCG, TD-Paint, and HiDiffusion, HRSino consistently achieves higher SSIM and PSNR across mask ratios, with improvements up to +0.03 SSIM and +1.8 dB PSNR. Fig 3 visualizes sinogram completion and reconstructed images, where HRSino produces nearly indistinguishable results from RePaint. These findings confirm that HRSino fundamentally extends diffusion-based completion to 2048×2048 resolution in a more memory- and runtime-efficient manner.

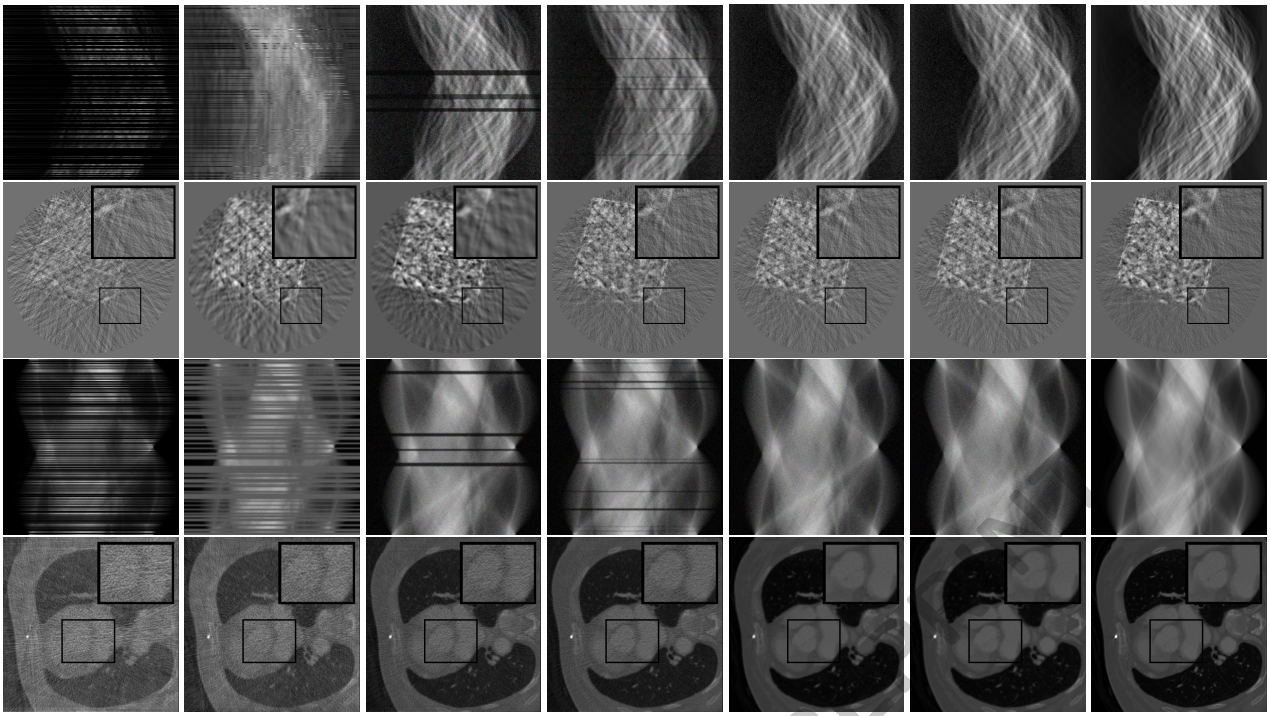

Figure 3: Qualitative completion results on TomoBank (lines 1 to 2) and LoDoPaB (lines 3 to 4) with random mask (ratio = 0.8) at 1024×1024 resolution. Odd columns and even columns show the sinograms and reconstructed images, respectively.

Method	Mem	Inf time	SSIM	PSNR
HRSino	24.7	2.25	0.927	30.6
W/o low	24.7	2.15	0.901	28.7
W/o mid	15.0	1.80	0.893	28.2
W/o high	24.5	1.20	0.870	27.1
W/o adaptive steps	24.7	3.05	0.928	30.9
W/o patch skipping	24.7	3.25	0.929	30.9

Table 3: Ablation study of multi-resolutions, adaptive denoising, and patch skipping. Results are reported as peak GPU memory (GB), inference time (s), and SSIM/PSNR on TomoBank with random masks (ratio = 0.8) at resolution 2048×2048.

Method	Mem	Inf time	SSIM	PSNR
HiDiffusion	35.7	2.73	0.900	28.8
HiDiffusion+HRSino	20.0	1.85	0.899	28.8
DiffIR	OOM			
DiffIR+HRSino	24.5	2.0	0.883	28.6

Table 4: Orthogonality study on TomoBank with random masks (ratio = 0.8) at resolution 2048×2048.

4.3 Ablation Studies

To analyze the impact of individual components in HRSino, we conduct ablation studies on its three core mechanisms: multi-resolution progressive inference, adaptive denoising based on structural complexity, and sparse patch skipping based on spectral sparsity. Tab 3 summarizes the results. Removing any resolution stage substantially degrades completion quality, indicating that different resolutions provide complementary benefits and that all stages are necessary to maintain overall fidelity. Peak memory is dominated by the mid-resolution stage, since high-resolution inference is performed patch-wise and patches are processed sequentially. Removing adaptive denoising or patch skipping does not affect SSIM/PSNR, but results in significant efficiency loss: inference time increases by 35.6% without adaptive steps (all set to 50) and by 44.4% without patch skipping. Overall, these results demonstrate the complementary roles of three mechanisms in enabling efficient high-resolution completion.

4.4 Compatibility with existing optimizations

Our optimizations are orthogonal to existing efficiency-oriented diffusion designs. HRSino is built upon RePaint, but our optimizations can be applied to other diffusion architectures. We combine HRSino with two recent baselines, HiDiffusion and DiffIR. As shown in Tab 4, integrating HRSino further reduces peak memory and inference time, while maintaining comparable completion quality.

5 Conclusion

We present HRSino, a novel efficient high-resolution sinogram completion framework. HRSino integrates mechanisms that adaptively allocate computation across resolutions, spatial regions, and inference depth. This design enables completion on 2048×2048 sinograms using a single GPU, significantly reducing memory usage and runtime while maintaining the completion fidelity.

Acknowledgments

The authors would like to thank the anonymous reviewers for their constructive comments and suggestions. This work was supported in part by the National Science Foundation (NSF) under awards CCF-2047516 (CAREER), CCF-2146873, CNS-2230944, IIS-2451436, and OAC-2403088, National Institutes of Health (NIH) under the award of R01 AG085616, Commonwealth Cyber Initiative (CCI) under the award HC-2Q26-032, and the U.S. Department of Energy (DOE) under Contract No. DE-AC02-06CH11357, including funding from the XSCOPE project and Laboratory Directed Research and Development Program (Project No. 2023-0104) at Argonne National Laboratory (ANL). It used resources of the Argonne Leadership Computing Facility (ALCF) and the Advanced Photon Source (APS), both DOE Office of Science user facilities at ANL, and was based on research supported by the U.S. DOE Office of Science ASCR Program under Contract No. DE-AC02-06CH11357. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funding agencies.

References

[Armato III *et al.*, 2011] Samuel G Armato III, Geoffrey McLennan, Luc Bidaut, Michael F McNitt-Gray, Charles R Meyer, Anthony P Reeves, Binsheng Zhao, Denise R Aberle, Claudia I Henschke, Eric A Hoffman, et al. The lung image database consortium (lidc) and image database resource initiative (idri): a completed reference database of lung nodules on ct scans. *Medical physics*, 38(2):915–931, 2011.

[Avrahami *et al.*, 2022] Omri Avrahami, Dani Lischinski, and Ohad Fried. Blended diffusion for text-driven editing of natural images. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18208–18218, 2022.

[Avrahami *et al.*, 2023] Omri Avrahami, Ohad Fried, and Dani Lischinski. Blended latent diffusion. *ACM transactions on graphics (TOG)*, 42(4):1–11, 2023.

[Chen *et al.*, 2016] Tianqi Chen, Bing Xu, Chiyuan Zhang, and Carlos Guestrin. Training deep nets with sublinear memory cost. *arXiv preprint arXiv:1604.06174*, 2016.

[Chung *et al.*, 2022] Hyungjin Chung, Byeongsu Sim, Dohoon Ryu, and Jong Chul Ye. Improving diffusion models for inverse problems using manifold constraints. *Advances in Neural Information Processing Systems*, 35:25683–25696, 2022.

[De Carlo *et al.*, 2018] Francesco De Carlo, Doğa Gürsoy, Daniel J Ching, K Joost Batenburg, Wolfgang Ludwig, Lucia Mancini, Federica Marone, Rajmund Mokso, Daniël M Pelt, Jan Sijbers, et al. Tomobank: a tomographic data repository for computational x-ray science. *Measurement Science and Technology*, 29(3):034004, 2018.

[E *et al.*, 2025] Jiaze E, Zhengchun Liu, Tekin Bicer, Srutarshi Banerjee, Rajkumar Kettimuthu, Bin Ren, and Ian T Foster. Sinotx: A transformer-based model for sinogram inpainting. *Electronic Imaging*, 37:1–6, 2025.

[Guo *et al.*, 2025] Jiaqi Guo, Santiago Lopez-Tapia, and Aggelos K Katsaggelos. Advancing limited-angle ct reconstruction through diffusion-based sinogram completion. *arXiv preprint arXiv:2505.19385*, 2025.

[Gürsoy *et al.*, 2014] Doga Gürsoy, Francesco De Carlo, Xianghui Xiao, and Chris Jacobsen. Tomopy: a framework for the analysis of synchrotron tomographic data. *Journal of synchrotron radiation*, 21(5):1188–1193, 2014.

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

[Jain *et al.*, 2020] Paras Jain, Ajay Jain, Aniruddha Nrusimha, Amir Gholami, Pieter Abbeel, Joseph Gonzalez, Kurt Keutzer, and Ion Stoica. Checkmate: Breaking the memory wall with optimal tensor rematerialization. *Proceedings of Machine Learning and Systems*, 2:497–511, 2020.

[Jin *et al.*, 2017] Kyong Hwan Jin, Michael T McCann, Emmanuel Froustey, and Michael Unser. Deep convolutional neural network for inverse problems in imaging. *IEEE transactions on image processing*, 26(9):4509–4522, 2017.

[Kalender, 2011] Willi A Kalender. *Computed tomography: fundamentals, system technology, image quality, applications*. John Wiley & Sons, 2011.

[Lee *et al.*, 2018] Hoyeon Lee, Jongha Lee, Hyeongseok Kim, Byungchul Cho, and Seungryong Cho. Deep-neural-network-based sinogram synthesis for sparse-view ct image reconstruction. *IEEE Transactions on Radiation and Plasma Medical Sciences*, 3(2):109–119, 2018.

[Leuschner and others, 2025] Johannes Leuschner et al. Deep inversion validation library. https://github.com/jleuschn/dival, 2025.

[Leuschner *et al.*, 2021] Johannes Leuschner, Maximilian Schmidt, Daniel Otero Baguer, and Peter Maass. Lodopab-ct, a benchmark dataset for low-dose computed tomography reconstruction. *Scientific Data*, 8(1):109, 2021.

[Li *et al.*, 2023] Yanyu Li, Huan Wang, Qing Jin, Ju Hu, Pavlo Chemerys, Yun Fu, Yanzhi Wang, Sergey Tulyakov, and Jian Ren. Snapfusion: Text-to-image diffusion model on mobile devices within two seconds. *Advances in Neural Information Processing Systems*, 36:20662–20678, 2023.

[Lu *et al.*, 2022] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. *Advances in Neural Information Processing Systems*, 35:5775–5787, 2022.

[Lugmayr *et al.*, 2022] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF*

conference on computer vision and pattern recognition, pages 11461–11471, 2022.

[Mayet *et al.*, 2025] Tsiry Mayet, Pourya Shamsolmoali, Simon Bernard, Eric Granger, Romain Hérault, and Clément Chatelain. Td-paint: Faster diffusion inpainting through time-aware pixel conditioning. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025.

[Meng *et al.*, 2023] Chenlin Meng, Robin Rombach, Ruiqi Gao, Diederik Kingma, Stefano Ermon, Jonathan Ho, and Tim Salimans. On distillation of guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14297–14306, 2023.

[Mohan *et al.*, 2015] K. Aditya Mohan, SV Venkatakrishnan, John W Gibbs, Emine Begum Gulsoy, Xianghui Xiao, Marc De Graef, Peter W Voorhees, and Charles A Bouman. Timbir: A method for time-space reconstruction from interlaced views. *IEEE Transactions on Computational Imaging*, 1(2):96–111, 2015.

[Pelt and Batenburg, 2013] Daniel Maria Pelt and Kees Joost Batenburg. Fast tomographic reconstruction from limited data using artificial neural networks. *IEEE Transactions on Image Processing*, 22(12):5238–5251, 2013.

[Saharia *et al.*, 2022] Chitwan Saharia, William Chan, Huiwen Chang, Chris Lee, Jonathan Ho, Tim Salimans, David Fleet, and Mohammad Norouzi. Palette: Image-to-image diffusion models. In *ACM SIGGRAPH 2022 conference proceedings*, pages 1–10, 2022.

[Salimans and Ho, 2022] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. *arXiv preprint arXiv:2202.00512*, 2022.

[Shannon, 1948] Claude E Shannon. A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423, 1948.

[Slaney and Kak, 1988] Malcolm Slaney and AC Kak. *Principles of computerized tomographic imaging*. IEEE press, 1988.

[Sohl-Dickstein *et al.*, 2015] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. pmlr, 2015.

[Song *et al.*, 2020] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

[Tumanyan *et al.*, 2023] Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to-image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1921–1930, 2023.

[Wan *et al.*, 2021] Ziyu Wan, Jingbo Zhang, Dongdong Chen, and Jing Liao. High-fidelity pluralistic image completion with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4692–4701, 2021.

[Wang *et al.*, 2004] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[Xia *et al.*, 2021] Wenhan Xia, Hongxu Yin, Xiaoliang Dai, and Niraj K Jha. Fully dynamic inference with deep neural networks. *IEEE Transactions on Emerging Topics in Computing*, 10(2):962–972, 2021.

[Xia *et al.*, 2023] Bin Xia, Yulun Zhang, Shiyin Wang, Yitong Wang, Xinglong Wu, Yapeng Tian, Wenming Yang, and Luc Van Gool. Diffir: Efficient diffusion model for image restoration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13095–13105, 2023.

[Xiao *et al.*, 2024] Guangxuan Xiao, Tianwei Yin, William T Freeman, Frédo Durand, and Song Han. Fastcomposer: Tuning-free multi-subject image generation with localized attention. *International Journal of Computer Vision*, pages 1–20, 2024.

[Yuan *et al.*, 2021] Geng Yuan, Xiaolong Ma, Wei Niu, Zhengang Li, Zhenglun Kong, Ning Liu, Yifan Gong, Zheng Zhan, Chaoyang He, Qing Jin, et al. Mest: Accurate and fast memory-economic sparse training framework on the edge. *Advances in Neural Information Processing Systems*, 34:20838–20850, 2021.

[Zhang *et al.*, 2023a] Guanhua Zhang, Jiabao Ji, Yang Zhang, Mo Yu, Tommi S Jaakkola, and Shiyu Chang. Towards coherent image inpainting using denoising diffusion implicit models. 2023.

[Zhang *et al.*, 2023b] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3836–3847, 2023.

[Zhang *et al.*, 2024a] Shen Zhang, Zhaowei Chen, Zhenyu Zhao, Yuhao Chen, Yao Tang, and Jiajun Liang. Hidiffusion: Unlocking higher-resolution creativity and efficiency in pretrained diffusion models. In *European Conference on Computer Vision*, pages 145–161. Springer, 2024.

[Zhang *et al.*, 2024b] Yang Zhang, Er Jin, Yanfei Dong, Ashkan Khakzar, Philip Torr, Johannes Stegmaier, and Kenji Kawaguchi. Effortless efficiency: Low-cost pruning of diffusion models. *arXiv preprint arXiv:2412.02852*, 2024.

[Zhu *et al.*, 2024] Haowei Zhu, Dehua Tang, Ji Liu, Mingjie Lu, Jintu Zheng, Jinzhang Peng, Dong Li, Yu Wang, Fan Jiang, Lu Tian, et al. Dip-go: A diffusion pruner via few-step gradient optimization. *Advances in Neural Information Processing Systems*, 37:92581–92604, 2024.