SmokeSVD: Smoke Reconstruction from A Single View via Progressive Novel View Synthesis and Refinement with Diffusion Models

Supplementary Material

A. Overview

In this supplementary material, we provide additional background, detailed descriptions of the technical approach, implementation specifics, evaluation results, and ablation studies. We also discuss the limitations of our work and outline potential directions for future research.

B. Preliminary

Navier-Stokes Equation. Generally, fluid motion is governed by the well-known incompressible Navier-Stokes equations:

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} = -\frac{\nabla p}{\rho} + \nu \nabla^2 \mathbf{u} + \mathbf{f}, \tag{11}$$

$$\nabla \cdot \mathbf{u} = 0, \tag{12}$$

where $\mathbf{u}$ is the velocity, ρ is the density, p is the pressure, $\mathbf{f}$ is the external force, and ν is the viscosity coefficient, which is usually set to zero for smoke phenomena. Eq. 11 is the momentum equation, which describes the time rate of velocity change, while Eq. 12 is the mass conservation equation to preserve the incompressibility. To formalize, density evolution follows the transport equation:

$$\frac{\partial \rho}{\partial t} + \mathbf{u} \cdot \nabla \rho = 0. \tag{13}$$

Diffusion Models. Diffusion probabilistic models (DDPM) consist of two processes: a forward diffusion process and a reverse inference process. During the training stage, given a data point $x_0 \sim q(x)$ sampled from the real data distribution, the forward process adds Gaussian noise to the sample x_0 over S time steps, constructing a Markov chain diffusion process:

$$q(x_s|x_{s-1}) = \mathcal{N}(x_s; \sqrt{1-\beta_s}x_{s-1}, \beta_s I), \tag{14}$$

$$q(x_{1:S}|x_0) = \prod_{s=1}^{S} q(x_s|x_{s-1}), \tag{15}$$

where $\mathcal{N}$ denotes a Gaussian distribution, β_s denotes a fixed or learnable variance schedule parameter that controls the noise intensity added at each step, x_s denotes the noisy image at time step s (selected from the total steps S), which can be expressed as:

$$x_s = \sqrt{\bar{\alpha}_s}x_0 + \sqrt{1-\bar{\alpha}_s}\epsilon, \tag{16}$$

Table S1. Key Mathematical Symbols

Symbol	Meaning
w_α^t	The smoke image at the tth frame and α viewing angle
$w_{c,\alpha}^t$	The clean image
$w_{r,\alpha}^t$	The rendered result for reconstructed density field
$w_{f,\alpha}^t$	The refined image
α	$\angle 0°$ for the input front view, $\angle 90°$ for the side view
I^t	The set of images from multiple views at the tth frame
ρ	Density field
$\hat{\rho}$	Advected density field
$\rho_{r,c}$	Coarse-grained reconstructed density field
$\rho_{r,f}$	Fine-grained reconstructed density field
$\mathbf{u}$	Velocity field
$\mathbf{u_r}$	Reconstructed velocity field
ρ_{in}	Inflow state
$\mathcal{A}$	Differentiable advection operator
$\mathcal{R}$	Differentiable rendering operator
SvDiff	Side-view synthesizer based on diffusion models
NvRef	Novel refinement module
$\mathcal{G}_\rho^c$	Coarse-grained density generator
$\mathcal{G}_\rho^f$	Fine-grained density generator
$\mathcal{G}_u$	Velocity generator

where $\alpha_s = 1 - \beta_s$, $\bar{\alpha}_s := \prod_{s=1}^{s} \alpha_s$, and $\epsilon \sim \mathcal{N}(0, I)$. The model is trained to minimize the following loss function:

$$\|\epsilon - \epsilon_\theta(x_s, s)\|^2. \tag{17}$$

During the generation stage, the diffusion model samples a Gaussian random noise $x_S \sim \mathcal{N}(0, I)$, and utilizes the predefined variance σ_s and random noise ϵ_s to gradually denoise it to until x_0. This process is formulated as:

$$\begin{aligned} x_{s-1} = \sqrt{\bar{\alpha}_{s-1}}\Big(\frac{x_s - \sqrt{1-\bar{\alpha}_s}\epsilon_\theta^{(s)}(x_s)}{\bar{\alpha}_t}\Big) \\ + \sqrt{1-\bar{\alpha}_{s-1}-\sigma_s^2}\cdot\epsilon_\theta^{(s)} + \sigma_s\epsilon_s, \end{aligned} \tag{18}$$

where $s = S, ..., 1$, and ϵ_θ is estimated noise from x_s.

C. Technical Details

C.1 Mathematical Symbols

Key mathematical symbols used in the paper are documented in Table S1.

C.2 Multi-frame Training Algorithm

If the previously synthesized frame is not used as one of the input conditions, the generated results exhibit significant cumulative errors, as shown in Fig. S1. To address this issue, we propose a multi-frame training algorithm, summarized in Alg. S1, which incorporates the estimated clean image from the previous time step as a conditional input for the subsequent forward diffusion process.

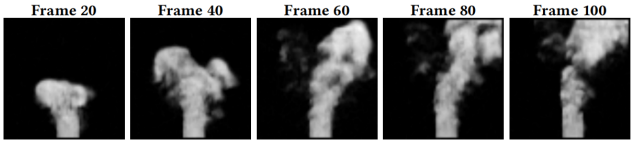

Figure S1. Side-view generation results affected by cumulative error.

Algorithm S1 Multi-frame Training Algorithm for SvDiff.

Require: Number of iterations it, noise steps S, noise threshold TQ

repeat
  Sample $s \sim \text{Uniform}(\{1, \dots, S\})$
  $\rho^{t-1} = \mathcal{G}_\rho(w^{t-1}_{\angle 0^\circ}, w^{t-1}_{\angle 90^\circ})$
  for $i = 0, 1, 2, \dots, it$ **do**
    Condition c^i : $w^{i+t-2}_{c,\angle 90^\circ}$, $w^{i+t-2}_{r,\angle 90^\circ}$, $w^{i+t-1}_{c,\angle 90^\circ}$, $w^{i+t-1}_{r,\angle 90^\circ}$, $w^{i+t}_{\angle 0^\circ}$
    Clean image sample $x^i_0 : w^{i+t}_{\angle 90^\circ}$
    Sample $\epsilon \sim \mathcal{N}(0, I)$
    $x^i_s = \sqrt{\bar{\alpha}_s} x^i_0 + \sqrt{1-\bar{\alpha}_s}\epsilon$
    $\hat{\epsilon} = \epsilon_\theta(x^i_s, c^i, s)$
    $\mathcal{L}_{noise} = \|\epsilon - \hat{\epsilon}\|^2$
    if $s < TQ$ **then**
      $\hat{x}^i_0 = \dfrac{x^i_s - \sqrt{1-\bar{\alpha}_s}\hat{\epsilon}}{\sqrt{\bar{\alpha}_s}}$
      $\rho^i_{r,c} = \mathcal{G}_\rho(w^{i+t}_{\angle 0^\circ}, w^{i+t}_{c,\angle 90^\circ})$
      $\mathbf{u}^{i-1} = \mathcal{G}_u(\rho^{i-1}, \rho^i_{r,c})$
      $w^{i+t}_{c,\angle 90^\circ} = \hat{x}^i_0, w^{i+t}_{r,\angle 90^\circ} = \mathcal{R}(\rho^i_{r,c}), \rho^{i-1} = \rho^i_{r,c}$
      $\mathcal{L}_{img} = \|x^i_0 - \hat{x}^i_0\|^2$
      $\mathcal{L}_{vel} = \|\nabla \cdot \mathbf{u}^{i-1}\|^2 + \|\nabla \mathbf{u}^{i-1}\|^2$
      $\mathcal{L}_{sp} = \|H(w^{i+t}_{c,\angle 90^\circ}) - H(w^{i+t}_{\angle 0^\circ})\|^2$
    else
      break
    end if
  end for
  Take gradient step on $\mathcal{L}_{SvDiff}$
until converged

Algorithm S2 Progressive Novel View Refinement.

Require: Current frame t; coarse density ρ^t_c; near/mid/far view sets nv, mv, fv; angular offset β; refined images from previous frames w^{t-1}_f, w^{t-2}_f

ViewSets $\leftarrow \{nv, mv, fv\}$
for each view set V in *ViewSets* **do**
  # Rendering and refinement for the same view type
  for each view angle α in V **do**
    $w^t_{r,\alpha} = \mathcal{R}(\rho^t_c, \alpha)$
    $w^t_{r,\alpha-\beta} = \mathcal{R}(\rho^t_c, \alpha - \beta)$
    $w^t_{r,\alpha+\beta} = \mathcal{R}(\rho^t_c, \alpha + \beta)$
    $w^t_{f,\alpha} = \text{NvRef}\Big(w^t_{r,\alpha-\beta} \oplus w^t_{r,\alpha+\beta} \oplus w^t_{r,\alpha}$
      $\oplus \downarrow w^{t-1}_{f,\alpha} \oplus \downarrow w^{t-2}_{f,\alpha}\Big)$
  end for
  # Density reconstruction using all refined images obtained
  $\rho^t_c = \mathcal{G}_\rho(\text{all refined imgs})$
end for
 # After the final iteration
$\rho^t_f \leftarrow \rho^t_c$

C.3 Progressive Refinement

As shown in Fig. S3, $\rho^t_{r,c}$ appears blurry in novel views due to limited available information. To address this, we introduce a progressive refinement module that incrementally enhances the blurred novel images, improving clarity from near to far views, as summarized in Alg. S2.

C.4 Density Generator

To provide 3D input from 2D images, we transform the image through expansion to match the required dimensions, and concatenate them from multiple viewpoints, as shown in Fig. S4. To be specific, $\mathcal{G}_\rho$ adopts the UNet3+ architecture with 3D convolutions.

C.5 Velocity Estimation

To reconstruct temporal and physically reasonable smoke dynamics, we establish a velocity generator $\mathcal{G}_u$ to estimate the velocity field based on two density fields of consecutive frames:

$$\mathbf{u}^t_r = \mathcal{G}_u(\rho^t, \rho^{t+1}), \tag{19}$$

which is supervised by $\mathcal{L}_u = \|\mathbf{u_r} - \mathbf{u}\|^2$. Additionally, to satisfy the divergence-free requirement in Eq. 12, we introduce another divergence loss as $\mathcal{L}_{div} = \|\nabla \cdot \mathbf{u_r} - \nabla \cdot \mathbf{u}\|^2$.

To ensure long-term robustness and reduce the adverse impact of the reconstruction errors in density, we employ a differentiable advection operator $\mathcal{A}$ based on Eq. 13, to formulate an advection loss term for the velocity generator. The advection operator $\mathcal{A}$ transports the density field ρ

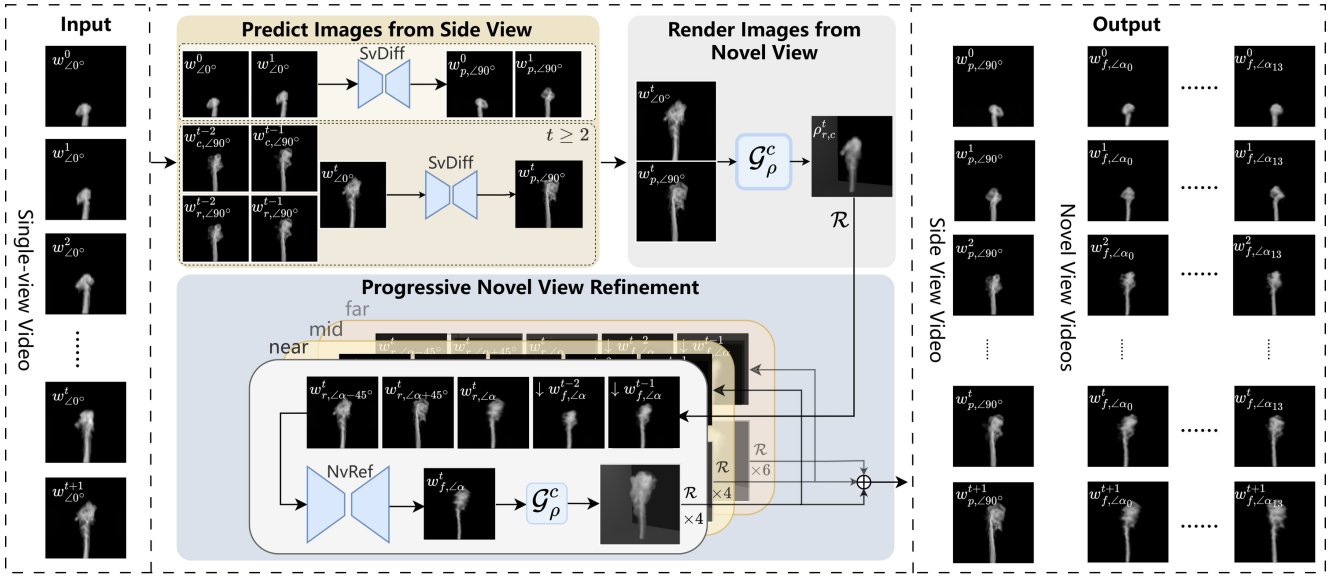

Figure S2. Procedure for side-view synthesis and novel view refinement. First, SvDiff predicts side-view images from input and previously generated images (when $t \geq 2$). Next, we reconstruct coarse density with $\mathcal{G}_\rho^c$ using front and side views, and render nearby novel views. Then, we iteratively refine novel views and reconstruct density, progressively extending from near to mid and far views, yielding multiple high-quality views for fine-grained reconstruction.

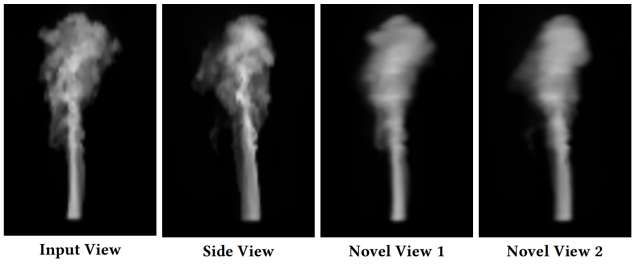

Figure S3. Rendering results of coarse-grained density field, which exhibits blurriness in novel views.

based on the velocity field $\mathbf{u}$, expressed as:

$$\hat{\rho}^t = \mathcal{A}(\rho^{t-1}, \mathbf{u}_r^{t-1}, \rho_{in}, dt), \tag{20}$$

where the density field obtained through velocity-based advection is called the advected density field, denoted as $\hat{\rho}$, ρ_{in} is the dynamic inflow, and dt is the time step. Similar to the density generator, we employ the following 3D density-based and 2D image-based advection loss terms:

$$\mathcal{L}_{advect} = \lambda_{\hat{\rho}} \|\rho - \hat{\rho}\|^2 + \lambda_{\mathcal{R}} \|\mathcal{R}(\rho) - \mathcal{R}(\hat{\rho})\|^2. \tag{21}$$

Based on the advected density field $\hat{\rho}$, we modify the input of $\mathcal{G}_u$ to ensure that the velocity field can be corrected through the advected density field, with the formula being:

$$\mathbf{u}_r^t = \mathcal{G}_u(\hat{\rho}^t, \rho^{t+1}). \tag{22}$$

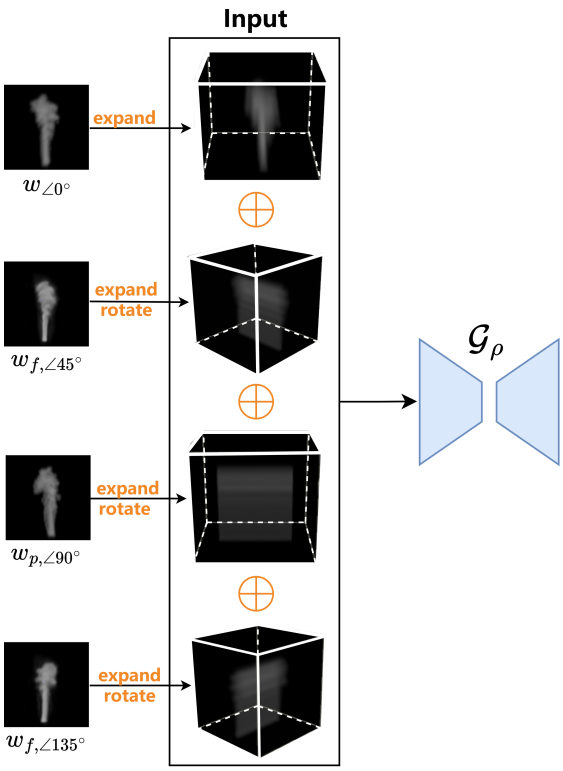

Figure S4. The architecture of density generator. The illustration depicts the case with four input images.

C.6 Inflow Estimation

The inflow state has a tremendous impact on the visual pattern of smoke phenomena, which cannot be ignored in smoke reconstruction. In long-term evolution, underestimating the inflow will lead to an inability to fill the smoke volume in later time steps, while overestimating can cause obvious instability, ultimately failing to match the input images [7].

To address this issue, we propose to estimate the inflow state frame-by-frame, that determines the inflow of current frame based on two adjacent density fields $\hat{\rho}^t$ and ρ^{t+1}, the velocity field $\mathbf{u}^t$, and the input image $w^{t+1}_{\angle 0^\circ}$. Specifically, for each frame, we initialize a random smoke source ρ_{in} and iteratively optimize the inflow source by minimizing the following loss function:

$$\begin{aligned}\mathcal{L}_s = &\|\rho_r^{t+1} - \mathcal{A}(\hat{\rho}_r^t, \mathbf{u}_r^t, \rho_{in}^t, dt)\|^2 \\ &+ \|w^{t+1}_{\angle 0^\circ} - \mathcal{R}(\mathcal{A}(\hat{\rho}_r^t, \mathbf{u}_r^t, \rho_{in}^t, dt), \angle 0^\circ)\|^2 \\ &+ \|\rho_{in}^{t-1} - \rho_{in}^t\|^2.\end{aligned} \tag{23}$$

Additionally, to prevent overestimation of the inflow source, we enforce zeroing out portions of the source that exceed a height threshold.

By incorporating the velocity and inflow estimation with density evolution [28], we can impose strong physical constraints to augment the temporal coherence and visual realism of SmokeSVD, thus effectively removing long-term flickers and non-physical artifacts in reconstructed smoke dynamics.

D. Implementation Details and Experimental Settings

Implementation Details. Our method is trained in two stages. In the first stage, we train SvDiff and NvRef based on the multi-frame training scheme to estimate clean images. We employ DDIM (Denoising Diffusion Implicit Models) sampling [34] described in Eq. 18 to accelerate the sampling process. Simultaneously, we also train the density generator $\mathcal{G}_\rho$ and the velocity generator $\mathcal{G}_u$. Our density generator $\mathcal{G}_\rho$ outputs smoke density fields with resolutions of 64^3 (for synthetic datasets) or $64 \times 112 \times 64$ (for real-world datasets). In the second stage, we fine-tune the velocity generator $\mathcal{G}_u$ based on the pre-trained density generator $\mathcal{G}_\rho$. All the aforementioned experiments were conducted on an NVIDIA GeForce RTX 3090 (24GB) GPU, while the performance was tested on an NVIDIA GeForce RTX 2080 Ti (11GB) GPU. Since optimization-based and neural radiance field (NeRF) methods require training for a few hours, far exceeding the minute-level time consumption of our proposed method, their specific time cost is not listed in the table.

Dataset. Based on the Eulerian method [21], we generated the required synthetic dataset by randomly modifying the wind fields, thermal fields, and the size and position of inflow regions in the scenarios. A total of 100 scenarios were generated, with each scene containing 150 frames. Additionally, we used post-processed images from the first 20 scenes of the ScalarFlow dataset [7] to train and evaluate our model.

Benchmarks. We compared our method with existing techniques that accept single-view videos as input for 3D smoke reconstruction, selecting GlobTrans [8], NGT [9], PICT [39], and PINF [5] as benchmarks. In our experiments, we modified the inputs of PICT and PINF to support single-view video input. Among these methods, GlobTrans reconstructs 3D smoke based on direct optimization algorithms, while PICT and PINF are based on Neural Radiance Fields (NeRF). These methods all require optimization for individual scenario, resulting in expensive time consumption and re-optimization requirement when changing scenarios. In contrast, the NGT method uses a trained neural network to estimate a single motion of smoke, avoiding direct optimization of the entire scenario, thereby significantly improving reconstruction speed and applicability.

Evaluation Metric. For image-related tasks (including novel view generation, refinement, and rendered images from reconstructed density fields), we use Mean Square Error (MSE), Root Mean Square Error (RMSE), Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM) [40], Fréchet Inception Distance (FID) [14], Learned Perceptual Image Patch Similarity (LPIPS) [53], and STYLE similarity to measure the similarity between generated images and ground truth images. The STYLE similarity is defined as the $L1$ difference between the Gram matrices of features extracted from the generated results and the ground truth using VGG19. Additionally, we evaluate the feature consistency between generated images and ground truth images with $\mathcal{L}_{sp}$. For reconstruction tasks, we use RMSE of density fields, divergence and gradient of velocity fields to measure the similarity between reconstructed and ground truth physical fields.

E. More Evaluations

Results on Synthetic Dataset. Fig. S5 demonstrates the qualitative performance of our method on the synthetic dataset, where the density field resolution of the reconstructed scenario is 64^3. By generating novel view images, our method significantly alleviates the ill-posed problem in single-view video based reconstruction, and the rendering results of reconstructed density fields perform well across different views.

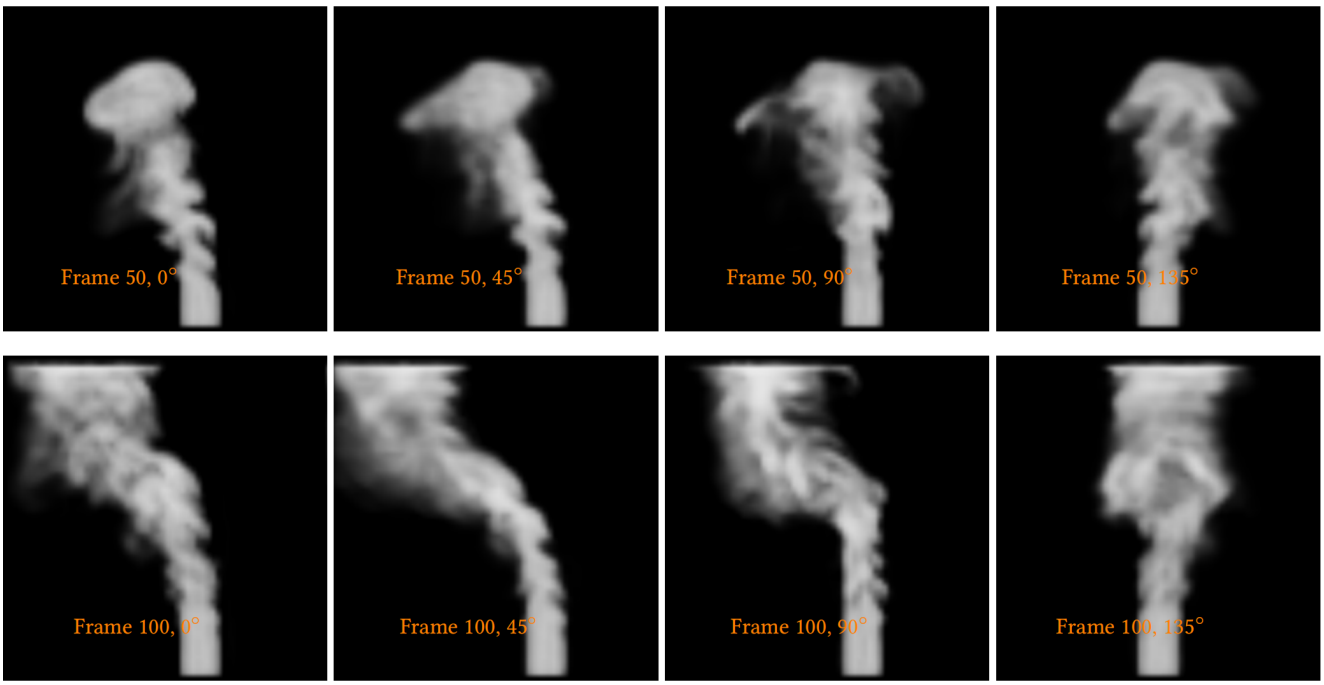

Figure S5. The rendering results of reconstructed density field at multiple views based on our proposed method.

Side-View Quality. We employ optical flow analysis as temporal consistency metrics (Table S2). We achieve performance closest to GT (15min vs. GT's 30 hours): Max (2nd best) indicates minimal flickers, Avg shows reasonable dynamics comparable to NGT/GT, and Std validates consistency. Note that PICT's low metrics stem from depth-blur eliminating motion detail.

Table S2. Optical flow statistics over 120 frames on ScalarFlow.

Metric	Reference	GT	NGT	PINF	PICT	FluidNexus	Ours
Max.	0.0896	**0.0953**	0.1272	0.1861	0.5890	0.2166	<u>0.1208</u>
Avg.	0.0593	<u>0.0639</u>	**0.0630**	0.1274	0.0253	0.0765	0.0767
Std Dev	0.0091	<u>0.0121</u>	0.0170	0.0185	**0.0116**	0.0348	0.0158

More Generalization Performance. We also test with multi-plume collisions and dry ice, as shown in Figs. S6 and S7. Our method performs well on various smoke shapes, which are fundamentally different from the single-source smoke scenes in our training dataset.

Interactive Simulation. Our reconstructed physical fields enable the re-simulation of input videos, and the generation of new smoke phenomena with controllable effects and enhanced detail, as shown in Figs. S8 and S9. In Fig. S9, we demonstrate re-simulation results in which a newly added spherical obstacle (top row) or external force field (bottom) is introduced by projecting the reconstructed velocity field onto a new simulation domain.

Compatibility with 3D Gaussian Splatting. Once sufficient novel views have been generated, our method can be seamlessly integrated with downstream applications such as 3D Gaussian Splatting (3DGS). As shown in Figs. S10 and S11, thanks to the multi-view consistency and well-structured spatiotemporal features provided by our approach, 3DGS is able to reproduce physically and visually plausible smoke sequences without the need for additional temporal processing.

F. More Ablation Studies

Effect of Frame Numbers. We adopted a multi-frame training strategy to train the side-view synthesizer (SvDiff) and the novel view refinement module (NvRef). Taking SvDiff as an example, in the early stages of training, We fed SvDiff one image for a single forward diffusion process; subsequently, we gradually increased the number of training frames and forward diffusion times until the synthesis quality met the expectation. To determine the final number of training frames and forward diffusion timesteps, we tested different hyperparameter settings for SvDiff. Since the number of training frames equals the number of forward diffusion times, we named these hyperparameter settings based on the number of frames (e.g., SvDiff-F1, SvDiff-F2), as shown in Fig. S12. As the number of training frames increased, the synthetic results gradually became more reasonable. For example, the SvDiff-F1 in Fig. S12 did not use the multi-frame information to estimate clean images,

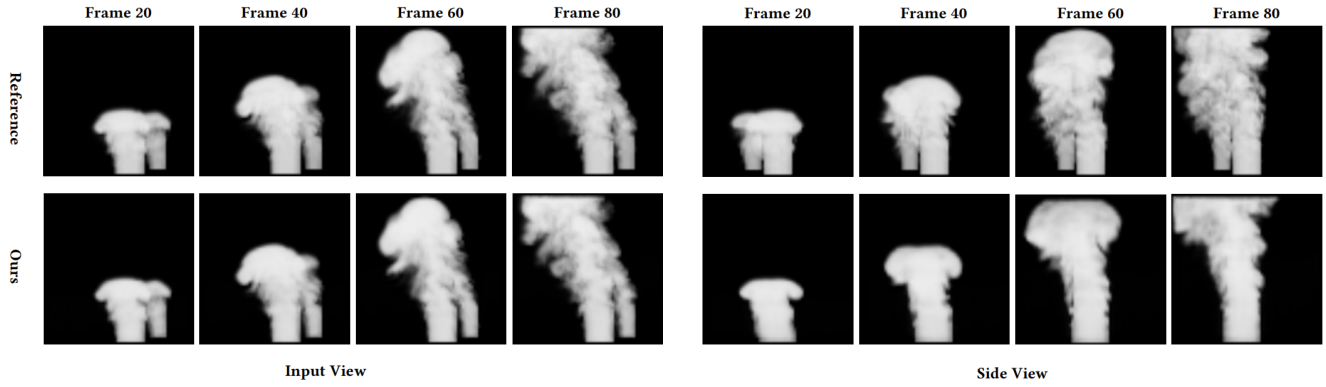

Figure S6. Reconstruction result for a multi-plume scenario, shown from both input and side views.

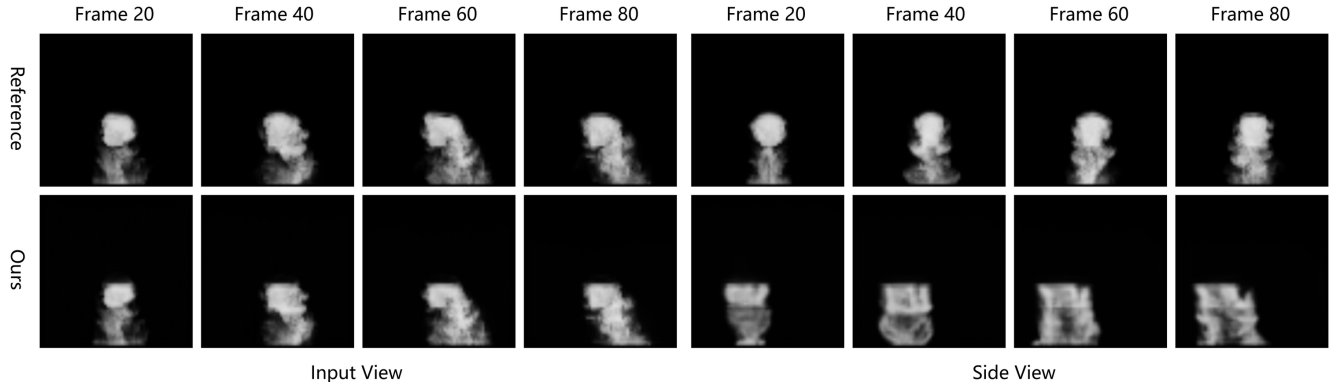

Figure S7. Reconstruction result for a dry ice scenario.

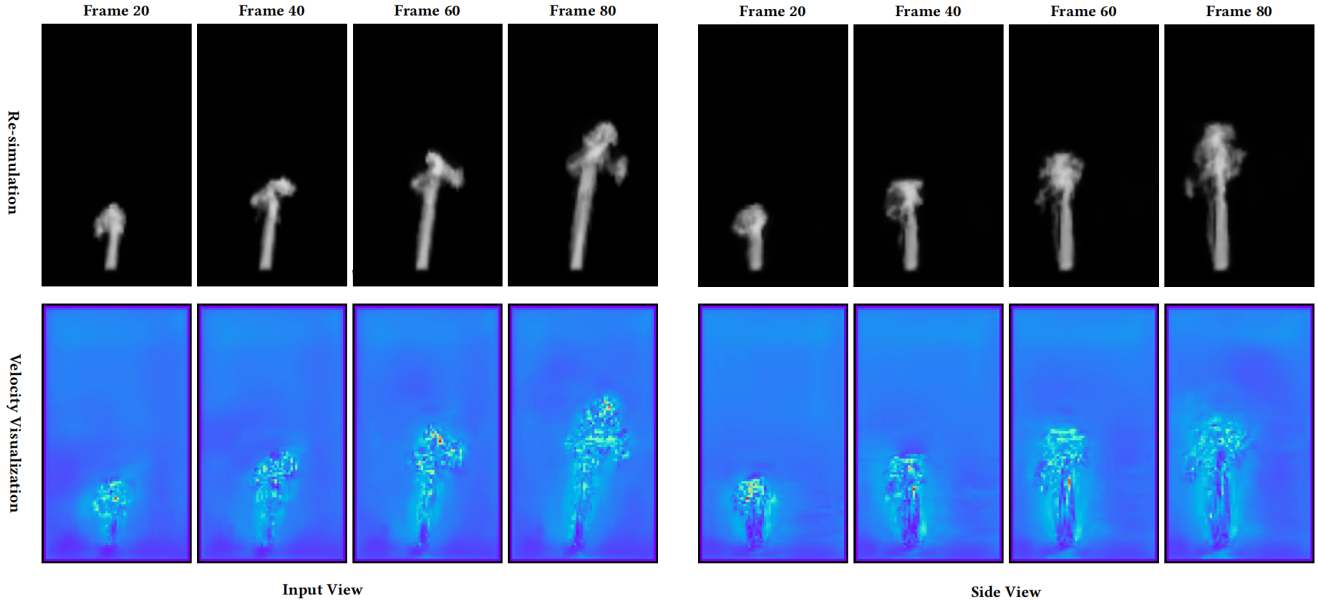

Figure S8. The rendered re-simulation results and velocity estimation visualization at the input view and the side view.

so due to the cumulative error, subsequent synthetic frames gradually deviated from reasonable smoke appearance. Ac-

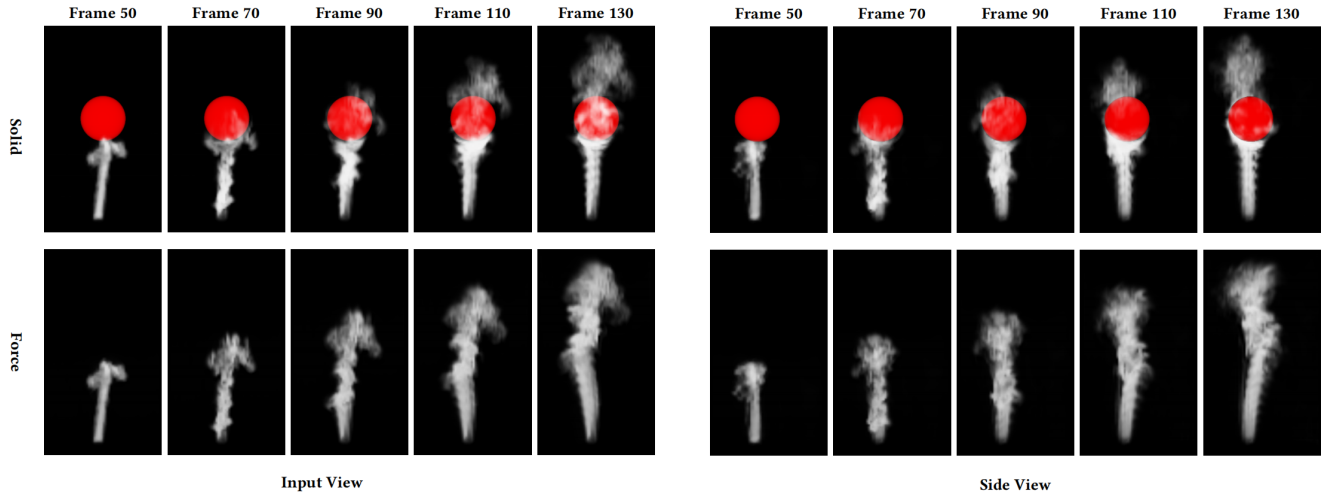

Figure S9. The re-simulation result with added fluid-solid coupling (top row), where we place a sphere obstacle (the red circle) at the 50th time step, and external force field (bottom row).

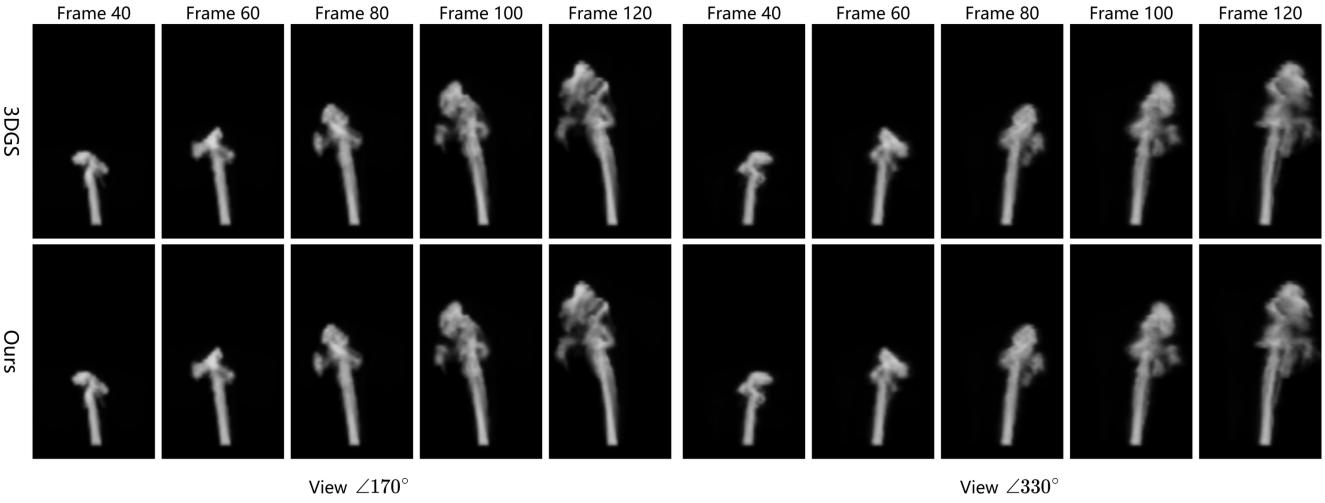

Figure S10. 3DGS results (top) based on our synthesized novel views (bottom).

cording to the results in Table S3, we found that the SvDiff based on four forward diffusions (SvDiff-F4) achieves the best. Both qualitative and quantitative evaluations indicate that the multi-frame training strategy based on estimated clean images plays a crucial role in the long-term generation process of diffusion models.

Effect of View Numbers. Our density generator can accept up to 16 smoke images from different viewpoints, with these views evenly distributed along a 180° arc. To determine the optimal number of input views for fine-grained density reconstruction, we trained several density generators using 2, 4, 8, and 16 input images (denoted as $2-, 4-, 8-, 16-\mathcal{G}\rho$), and evaluated their performance. The quantitative results are presented in Table S4. In the exper-

Table S3. Quantitative comparison of SvDiff with different frame numbers on the synthetic dataset. We report $\mathcal{L}_{sp}$, LPIPS, and SSIM to measure the differences between synthetic images and reference images, and warp error to measure pixel-level distortion between consecutive frames based on mean squared error (MSE).

Algorithm	$\mathcal{L}_{sp}\downarrow$	Warp Error↓	LPIPS↓	SSIM↑
reference	/	0.0981	/	/
SvDiff-F1	1.2601	0.2003	0.3873	0.4364
SvDiff-F2	1.2673	0.1819	0.3742	<u>0.5077</u>
SvDiff-F3	1.0422	**0.0915**	0.3910	0.4997
SvDiff-F4	**0.3475**	0.1481	**0.3384**	**0.5729**
SvDiff-F5	<u>0.7081</u>	<u>0.1259</u>	<u>0.3779</u>	0.5052

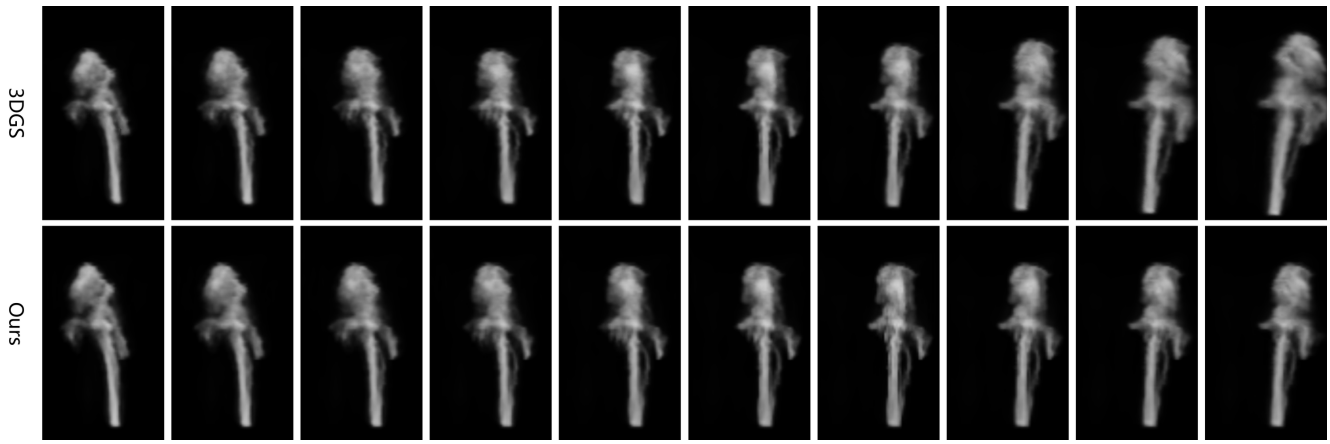

Figure S11. 3DGS results (top) and our reconstruction result (bottom) under rotating views from $\angle 210^\circ$ to $\angle 300^\circ$.

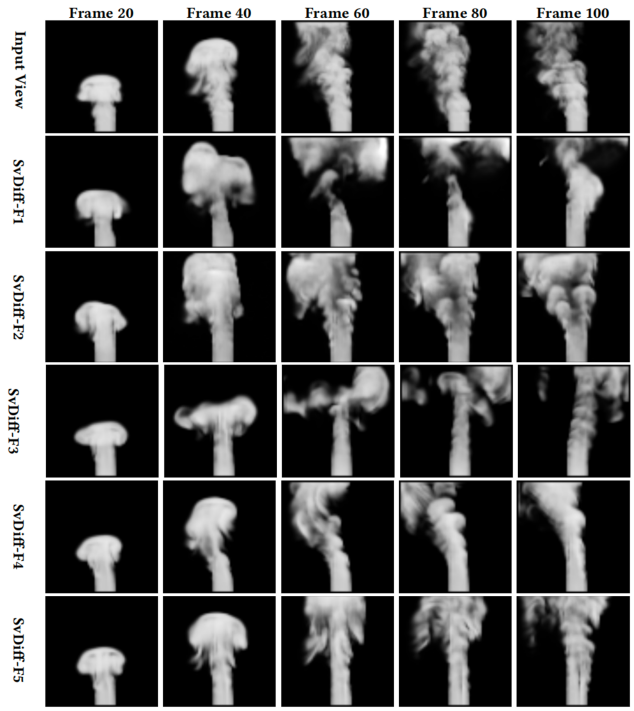

Figure S12. Qualitative comparison of side view synthesis with different frame numbers on the synthetic dataset.

iment, when the number of input images was less than 16, images from other novel views were masked. All image metrics were evaluated based on 16 real viewpoints, and the quantitative analysis indicates that as the number of input views increases, the reconstruction quality gradually improves. Therefore, in the coarse-grained density reconstruction stage, we used only a subset of views as input, whereas in the fine-grained stage, all 16 input views were utilized to provide richer information for high-quality reconstruction.

Table S4. Quantitative evaluation of density generators with different numbers of input views on the synthetic dataset. The last five metrics are evaluated based on images from 16 views.

View Num	ρ RMSE↓	RMSE ↓	SSIM↑	PSNR↑	LPIPS↓	FID↓
2	0.0356	0.0206	0.9795	37.0H561	0.0417	31.0919
4	0.0256	0.0100	0.9915	43.1682	0.0205	9.7665
8	<u>0.0186</u>	<u>0.0058</u>	<u>0.9960</u>	<u>47.2533</u>	<u>0.0099</u>	<u>2.5882</u>
16	**0.0148**	**0.0043**	**0.9974**	**49.6970**	**0.0050**	**1.3745**

Ablation on Side-view Synthesizer. We also visualized the maximum values and gradient of reconstructed velocity fields in Figs. S14 and S15.

Ablation on Key Components. Figs. S16 and S17 show NGT combined with our refinement and reconstruction. Our approach is compatible with NGT and further enhances its results, achieving high-quality reconstruction.

G. Limitation and Discussion

While our proposed framework demonstrates strong performance in reconstructing dynamic smoke from single-view input, several limitations remain. First, the current method assumes a relatively clean background and consistent lighting conditions; in real-world scenarios with complex backgrounds or varying illumination, the quality of side-view synthesis and subsequent reconstruction may degrade. Second, although our progressive refinement strategy improves multi-view consistency, the approach still relies on the accuracy of the initial side-view synthesis, significant errors in early stages can propagate and affect the final results. Third, our model is primarily evaluated on synthetic and controlled real-world datasets; its generalization to highly diverse or outdoor smoke phenomena remains to be further validated. Additionally, the computational cost, while lower than optimization-based methods, can still be significant

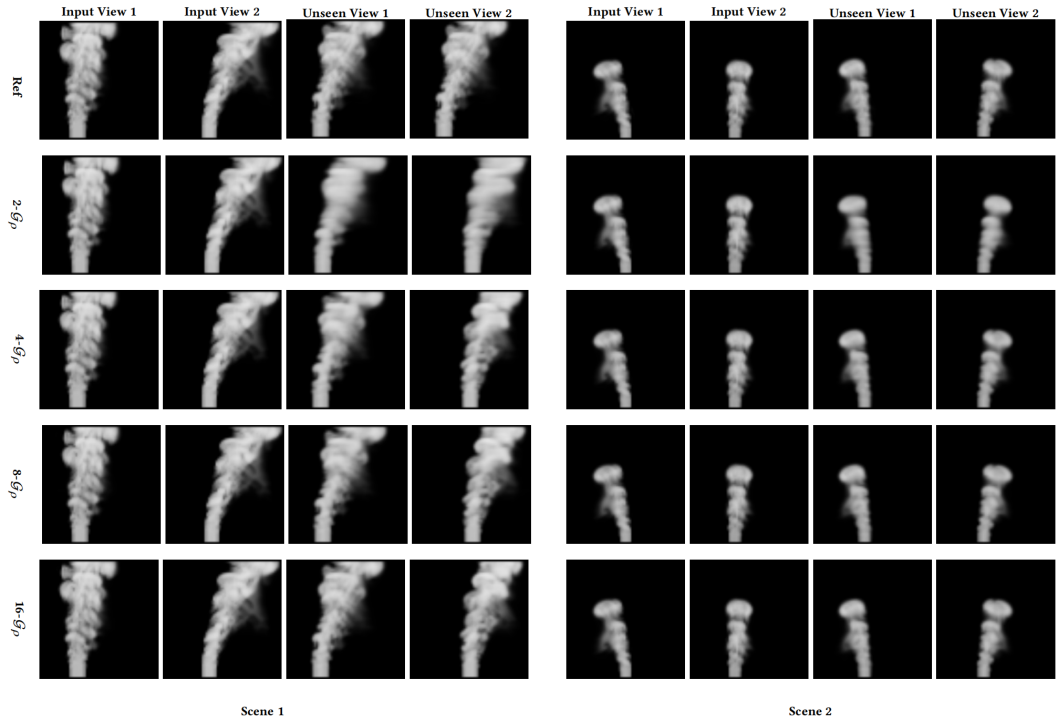

Figure S13. Qualitative comparison of density generators with different numbers of views on the synthetic dataset.

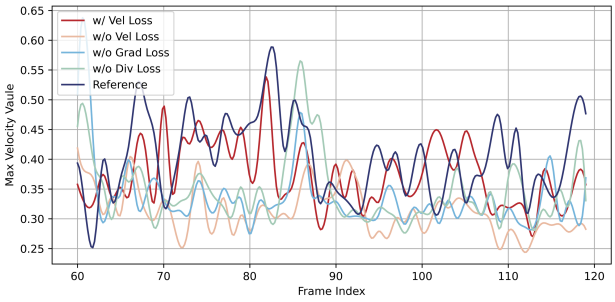

Figure S14. Comparison of the maximum values of reconstructed velocity fields by SvDiff with different loss functions at various time steps.

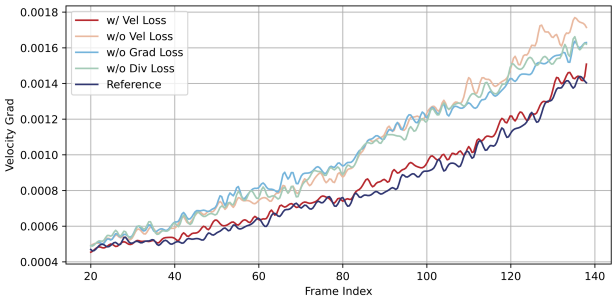

Figure S15. Comparison of the gradient of reconstructed velocity fields by SvDiff with different loss functions at various time steps.

when scaling to higher resolutions or longer sequences. Finally, our framework currently focuses on grayscale smoke and does not explicitly handle colored smoke, solid obstacles, or interactions with complex environments. Future work could address these limitations by incorporating more robust background modeling, exploring domain adaptation techniques, extending the framework to handle color and multi-phase flows, and integrating more advanced physical constraints to further enhance realism and generalization.

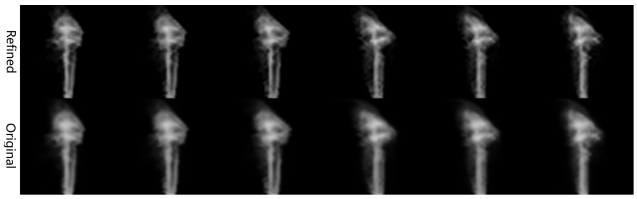

Figure S16. NGT combined with our refinement model.

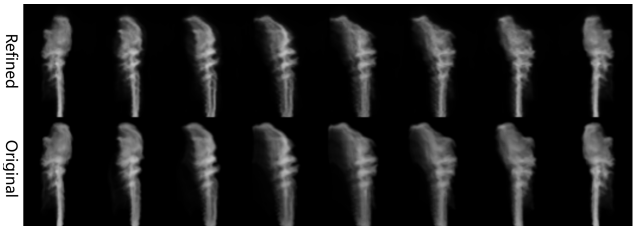

Figure S17. NGT combined with our reconstruction model.