2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

GAF: Gaussian Action Field as a 4D Representation for Dynamic World Modeling in Robotic Manipulation

Ying Chai* [1], Litao Deng* [2,3], Ruizhi Shao [1], Jiajun Zhang [1], Kangchen Lv [1], Liangjun Xing [1], Xiang Li† [1] Hongwen Zhang† [2], Yebin Liu† [1]
[1]Tsinghua University [2]Beijing Normal University [3]Shadow AI

***Abstract*— Accurate scene perception is critical for vision-based robotic manipulation. Existing approaches typically follow either a Vision-to-Action (*V-A*) paradigm, predicting actions directly from visual inputs, or a Vision-to-3D-to-Action (*V-3D-A*) paradigm, leveraging intermediate 3D representations. However, these methods often struggle with action inaccuracies due to the complexity and dynamic nature of manipulation scenes. In this paper, we adopt a *V-4D-A* framework that enables direct action reasoning from motion-aware 4D representations via a Gaussian Action Field (GAF). GAF extends 3D Gaussian Splatting (3DGS) by incorporating learnable motion attributes, allowing 4D modeling of dynamic scenes and manipulation actions. To learn time-varying scene geometry and action-aware robot motion, GAF provides three interrelated outputs: reconstruction of the current scene, prediction of future frames, and estimation of init action via Gaussian motion. Furthermore, we employ an action-vision-aligned denoising framework, conditioned on a unified representation that combines the init action and the Gaussian perception, both generated by the GAF, to further obtain more precise actions. Extensive experiments demonstrate significant improvements, with GAF achieving +11.5385 dB PSNR, +0.3864 SSIM and -0.5574 LPIPS improvements in reconstruction quality, while boosting the average +7.3% success rate in robotic manipulation tasks over state-of-the-art methods.**

I. Introduction

Effective perception is fundamental to robotic manipulation in unstructured 3D environments. Recent advances in vision-based methods have enabled robots to infer actions directly from visual observations by leveraging powerful foundation models [1], [2], [3], which facilitates the high-level scene understanding and robotic manipulation. Existing approaches for vision-based manipulation can be broadly categorized into two paradigms. ***V-A*** (vision-to-action) paradigm [4], [5], [6] directly map RGB observations to action sequences. While these methods benefit from end-to-end learning, they rely on implicit scene understanding and lack the modeling of 3D world. ***V-3D-A*** (vision-to-3D-to-action) paradigm [7], [8], [9] incorporate 3D representations such as point clouds [10], [11] and voxel grids [12], [13] to enable explicit geometric reasoning. By incorporating such structured 3D representations, V-3D-A methods can capture accurate spatial relationships within the scene, leading to higher success rates. Nonetheless, 3D-based methods often require 3D datasets which are much less available than image datasets. Despite their differences, both paradigms share a common limitation: they fail to capture the temporal evolution of scene geometry, leading to an inherent mismatch between static scene understanding and dynamic action generation.

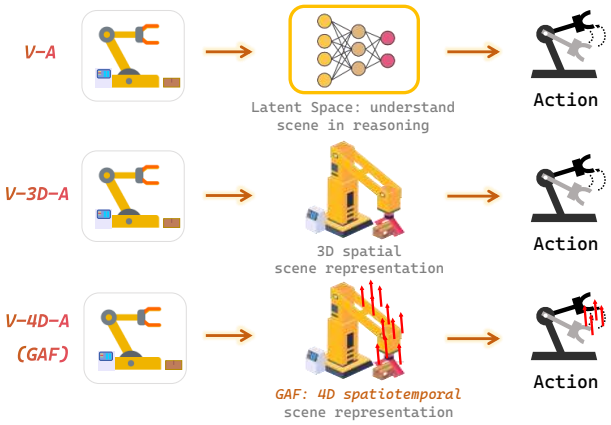

Fig. 1: Comparisons between the previous ***V-A*** paradigm, ***V-3D-A*** paradigm and our proposed ***V-4D-A*** method GAF.

In response to this challenge, the ***V-4D-A*** (vision-to-4D-to-action) paradigm has recently emerged, which augments 3D representations with motion information to capture the temporal evolution of scenes, as illustrated in Fig. 1. Unlike static representations that passively encode geometry, these methods aim to guide robotic action planning by modeling how scene geometry, including the robot itself, may evolve over time. Such 4D dynamic perception enable more intuitive action inference since the scene motion inherently contains the movement trend information.

Building on this paradigm, recent efforts [14], [15] have begun to explore 4D representations for robotic manipulation tasks. Most of these approaches model 4D representations primarily by leveraging the sequential nature of video frames, without explicitly embedding dynamic information into the scene representation. Some recent works address this limitation by integrating dynamic information more naturally into the scene perception through Gaussian-based world model: ManiGaussian [16] and GWM [17] deform current Gaussian for future scene consistency to further supervise action prediction. However, their methods only use the future states as volumetric priors to guide policy training. Their implicit use of Gaussian is less effective due

*Equal contributions. †Corresponding author

This work was supported by the National Natural Science Foundation of China (NSFC) No.62125107.

to the low fidelity. In this end, we seek to construct a more explicit 4D representation by directly modeling the temporal transformation between high-fidelity current and future states, which enables reliable action inference grounded in accurate dynamic scene understanding.

In this paper, we introduce the **Gaussian Action Field (GAF)** as a concrete implementation of the V-4D-A paradigm. GAF is built on 3D Gaussian Splatting (3DGS) [18] due to its strong geometric fidelity and its differentiable rendering mechanism, which allows supervision from RGB video frames without requiring ground-truth 3D data. To extend 3DGS to dynamic scenes, GAF introduces a learnable motion attribute, which encodes the temporal displacement of each Gaussian point to capture evolving scene geometry and infer motion within a unified representation. GAF produces three interrelated outputs, each corresponding to a key stage in the perception-to-action pipeline: The current Gaussian provides a view-consistent encoding of the present scene. The future Gaussian predicts how the scene evolves by applying the learned motion attributes. The init action is computed through point cloud registration between the current and future Gaussian. These components together allow GAF to connect dynamic visual perception with action generation, forming a complete V-4D-A framework. To further improve action quality, we introduce an action-vision-aligned denoising module that refines init actions using action-aware visual guidance produced by GAF. This overall approach resonates with the concept of the world model [19], modeling future scene dynamics to support downstream decision-making.

GAF operates in a fully feed-forward manner and supports real-time execution on a single GPU during manipulation. Extensive experiments demonstrate that our method enables high-quality scene reconstruction, accurate robotic manipulation and spatial generalization, outperforming V-A and V-3D-A baselines. Moreover, the approach has been successfully deployed in real-world environments, demonstrating its practical feasibility. Contributions are summarized as follows:

- We introduce a V-4D-A method GAF, extended 3DGS with motion inference, unifying dynamic scene evolution and future-oriented action prediction.
- We propose an action-vision-aligned denoising framework to enable action refinement within a unified image space, enhancing the accuracy of action prediction.
- We validate our method on robotic manipulation tasks, where it achieves state-of-the-art performance in both scene reconstruction quality and action generation.

II. Related Work

A. Vision-based Robot Learning

Vision [6], [10], [11], [20] plays a pivotal role in enabling robots to perceive and interact with their environments. 2D methods that rely on image inputs are the earliest to emerge and are supported by the most extensive datasets. Several of these methods have demonstrated remarkable performance. The Google RT series [21], IGOR [22] and ViLBERT [23] have achieved impressive results through massive-scale data training, while Diffusion Policy [4] and GENIMA [5] utilizes diffusion model to generate action sequence. These methods often struggle with accurately capturing precise 3D spatial relationships, limiting their effectiveness in high-precision tasks [24]. In contrast, 3D methods explicitly model volumetric geometric structures: Act3D [7] and RVT series [25] utilize point clouds for scene representation. Others, including PerAct [12], VoxAct [13] and GNFactor [9] adopt voxel grids to encode the scene geometry.These 3D representations enabling accurate spatial structure and higher success rate.

These methods neglect the fact that, in addition to complex geometric structures and spatial relationships, robot learning also requires the consideration of time as a crucial dimension. Unlike previous methods, our approach explicitly models temporal dynamics, allowing accurate 3D scene understanding and action prediction solely from RGB video frames.

B. World Model

World models obtain environmental knowledge by constructing an internal representation that simulates how the world evolves [19], [26], [27], [28]. These methods successfully encode scene dynamics [29], [30] by predicting future states from current observations. Previous approaches utilize auto-encoding to learn a latent space for predicting future states. The implicit nature of their feature representations, combined with high data requirements, limits the effectiveness and practical applicability of these methods. Recent methods enhance generalization by using explicit representations in image [31] and language [32], [33]: ManiGaussian [16] and GWM [17] adopt Gaussian as state representation and leverage predicted future Gaussian to guide policy training.

Unlike action-conditioned world models [16], [17] that use future prediction as auxiliary supervision, our GAF is vision-driven and task-oriented: it forecasts scene evolution directly from visual inputs and explicitly infers actions from the predicted future geometry, enabling direct action extraction and more efficient inference.

III. Method

In this section, we introduce GAF, its implementation, and its application in robotic manipulation tasks. Sec. III-A defines GAF representation and its outputs. Sec. III-B details GAF network design. Sec. III-C illustrates how the outputs of GAF are used to generate executable actions for a complete robotic manipulation task.

Our overall pipeline consists of two stages: 4D GAF scene reconstruction and action refinement via diffusion. In the first stage, GAF reconstructs dynamic scenes using only sparse multi-view RGB images and their corresponding intrinsics, without requiring camera extrinsics. In the second stage, the refinement module leverages both the reconstructed Gaussians and the known camera extrinsics (obtained via calibration) to render action-conditioned visual guidance for denoising.

A. GAF Representation

We define the Gaussian Action Field (GAF) as a unified spatiotemporal representation that associates each Gaussian

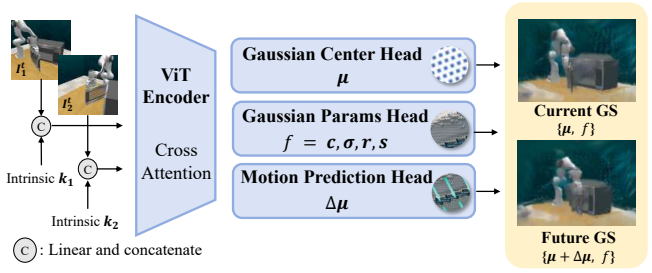

Fig. 2: **Overview of GAF reconstruction.** Given sparse multi-view images, a Vision Transformer extracts hybrid scene features, which are decoded by three heads to predict Gaussian positions, motions, and appearance parameters.

primitive $\mathbf{g}(\mathbf{x})$ at time step t with both geometric attributes and motion dynamics. Formally, GAF is parameterized by a continuous function $\mathcal{F}_\Theta$:

$$\mathcal{F}_\Theta : \{g(x), t\} \mapsto \{\mu, \Delta\mu, f\}, \tag{1}$$

where $\mu \in \mathbb{R}^3$ denotes the 3D position of the Gaussian, $\Delta\mu \in \mathbb{R}^3$ is displacement vector indicating temporal motion, and appearance parameters $f = \{c, \sigma, r, s\}$ represents the color, opacity, rotation, and scale attributes of each Gaussian.

GAF produces three types of outputs: the current Gaussian, the future Gaussian, and the init action. These outputs are generated from different combinations of parameters. The current Gaussian is constructed from the position and appearance parameters $\{\mu, f\}$, representing the present scene. The future Gaussian is obtained by applying the predicted motion to the position parameters and combining the result with the same appearance features, yielding $\{\mu + \Delta\mu, f\}$. The init action is estimated through point cloud matching between the current and future manipulation-related Gaussian.

B. GAF Architecture

The Gaussian Action Field (GAF) architecture unifies scene representation, scene dynamics, and action reasoning. Our goal is to reconstruct motion-augmented Gaussian directly from sparse, unposed RGB inputs, enabling downstream manipulation control. Fig. 2 illustrates the overall design.

a) Dynamic Gaussian Reconstruction: GAF adopts a geometry-agnostic, pose-free approach for dynamic scene reconstruction, in contrast to traditional methods such as NeRF [34] and 3DGS [18], which rely on dense camera poses or strong geometric priors like cost volumes, epipolar constraints. Our architecture directly reconstructs high-fidelity motion-augmented Gaussian by two input views in a canonical space aligned with the first input view. This is achieved using a feed-forward network that includes a vision transformer backbone and three specialized heads.

Specifically, given two unposed $H \times W$ images and their corresponding intrinsics $\{I_v^t, k_v^t\}_{v=1}^V$ at timestep t, we tokenize images into patch sequences and concatenate them. The resulting tokens are fed into a shared-weight Vision Transformer with cross-view attention to extract features.

For scene representation, we employ a decoupled two-head design $\mathcal{H}_{\text{Gauss}} = \{h_{\text{center}}, h_{\text{feature}}\}$ based on the DPT architecture[35] to process the features: the Gaussian Center Head h_{center} predicts only Gaussian point positions, the Gaussian Param Head h_{feature} estimates the remaining Gaussian parameters named as appearance parameters $f = \{c, \sigma, r, s\}$. The process can be formulated as:

$$\mathcal{H}_{\text{Gauss}}\left(\text{ViT}(\{\boldsymbol{I}_v^t, \boldsymbol{k}_v^t\})\right)_{v=1}^V = \{\mu_j^t, c_j^t, \sigma_j^t, r_j^t, s_j^t\}_{j=1}^{V\times H\times W}, \tag{2}$$

For scene dynamics, we introduce a Motion Prediction Head h_{motion} following the same DPT-based architecture[35] as Gaussian Center Head. h_{motion} predicts the per-point displacement $\Delta\mu_j^{t\to t+\Delta t}$, representing the motion of each Gaussian over a future interval Δt:

$$h_{\text{motion}}\left(\text{ViT}(\{\boldsymbol{I}_v^t, \boldsymbol{k}_v^t\})\right)_{v=1}^V = \{\boldsymbol{\Delta\mu}_j^{t\to t+\Delta t}\}. \tag{3}$$

The predicted displacement $\Delta\mu_j^{t\to t+\Delta t}$ are added to μ_j^t to obtain the future Gaussian positions $\mu_j^{t+\Delta t}$. These displaced centers are fused with the appearance parameters $(c_j^t, \sigma_j^t, r_j^t, s_j^t)$ to form the future Gaussian.

Deriving the current Gaussian and future Gaussian , we can render M novel view images for the current state $\{\hat{I}_v^t\}_{v=1}^M$ and future state $\{\hat{I}_v^{t+\Delta t}\}_{v=1}^M$ using alpha-blend rendering, M denote the number of synthesized views. To be specific, the pixel color at location $\mathbf{p}$ is computed by:

$$C(\mathbf{p}) = \sum_{i=1}^{N} \alpha_i c_i \prod_{j=1}^{i-1}(1-\alpha_j), \tag{4}$$

where C is the rendered image, N denotes the number of Gaussian, $\alpha_i = \sigma_i e^{-\frac{1}{2}(\mathbf{p}-\mu_i^{2d})^\top \Sigma_i^{-1}(\mathbf{p}-\mu_i^{2d})}$ represents the 2D density in the splatting process, and Σ_i stands for the covariance matrix acquired from the rotation r and scales s. M denote the number of synthesized views.

This allows for direct RGB video frames supervision for the entire Dynamic Gaussian Reconstruction. We learn $\mathcal{F}_\Theta$ by minimising the following [36]:

$$\mathcal{L}_{\text{GAF}} = \mathcal{L}_{\text{LPIPS}}^t + \mathcal{L}_{\text{MSE}}^t + \mathcal{L}_{\text{LPIPS}}^{t+\Delta t} + \mathcal{L}_{\text{MSE}}^{t+\Delta t}. \tag{5}$$

where $\mathcal{L}^t$ enforces geometric fidelity to current observations and $\mathcal{L}^{t+\Delta t}$ regularizes future state prediction. They are aggregated into a unified objective, facilitating the joint optimization of motion-augmented Gaussian reconstruction.

With access to 4D Gaussian-based perception at both the current and future time steps, we are able to extract more concrete actions from such representations.

b) Init Action Computation: Since our task centers on manipulation, our attention is directed toward the motion of the robotic arm itself, especially the end-effector. Due to its rigid nature, we extract the gripper-part Gaussians $\boldsymbol{\mu}_{\text{gripper}}$ and future state Gaussians $(\boldsymbol{\mu} + \boldsymbol{\Delta\mu})_{\text{gripper}}$, and estimate a rigid transformation $T^{t\to t+\Delta t} \in \text{SE}(3)$ using ICP [37]:

$$T^{t\to t+\Delta t} = \arg\min \sum_{k\in gripper} \|T(\mu_k) - (\boldsymbol{\mu} + \boldsymbol{\Delta\mu})_k\|^2 \tag{6}$$

$T^{t\to t+\Delta t}$ denotes the end-effector transformation matrix over a time interval Δt, providing an explicit representation

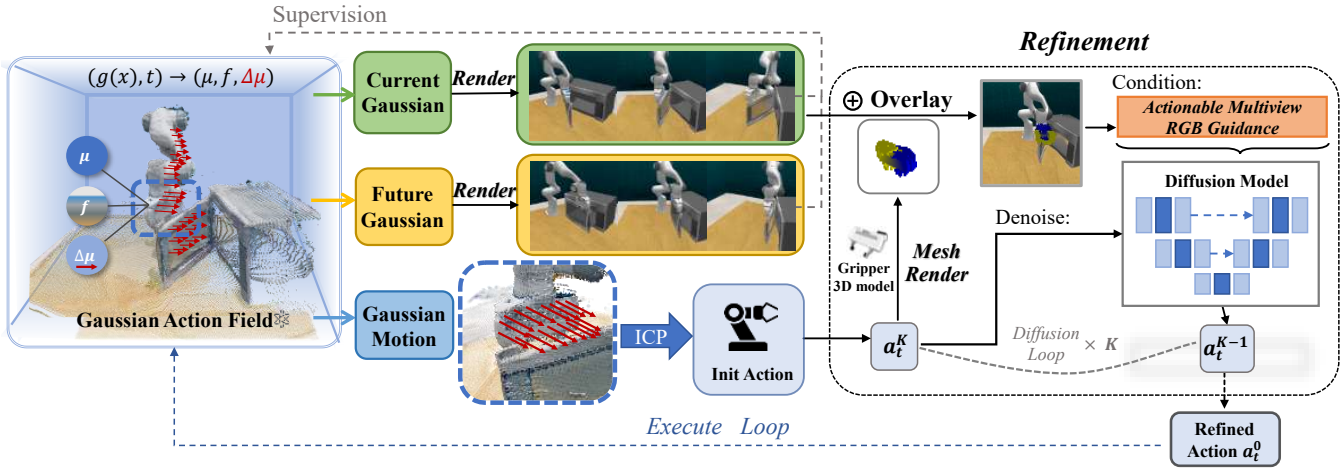

Fig. 3: **Manipulation pipeline.** GAF outputs are then used as conditions for a action-vision-aligned denoising framework to generate executable motion. The process repeats iteratively until the task completes.

of scene dynamics. We interpolate it into a sequence of transformations across discrete timesteps. This sequence forms the init action a_{init}, describing the transition from the current frame T to the future frame $t + \Delta t$.

C. Manipulation with GAF

After introducing the definition and architecture of GAF, we now describe how it is deployed in manipulation tasks. As the GAF module is trained solely on visual data without real action supervision, the predicted init actions are only visually plausible and often lack physical feasibility during interaction. To address this, we adopt a denoising network and incorporate a small amount of real action data to refine the initial predictions and ensure physically consistent execution.

As illustrated in Fig. 3, to fully exploit GAF's outputs, we adopt action-vision-aligned denoising framework inspired by R&D [38]. Specifically, we use a rendering process to visualize the spatial consequences of candidate actions. To determine the pose in which the gripper should be rendered, we first compute its pose in the world frame as $T^w_{g_{new}} = T^w_g \times a$, where a is a relative transformation representing the intended action. Then, by utilising the camera c's extrinsic matrix T_{w2c} and intrinsic matrix K, we can reposition the gripper's CAD model in the camera's frame and render an image of it, creating the rendered action representation R^c:

$$R^c = Render(T_{w2c} \times T^w_g \times a, K^c) \tag{7}$$

For each denoising step of duration Δt , we project the gripper positions resulting from the init action a_{init} to pixel coordinates and render gripper mesh onto current multi-view RGB images $\{\hat{I}^t_v\}^M_{v=1}$. These multiview rendered action representations R^c create a unified representation, termed *Actionable Multiview RGB Guidance*, which integrates the visual 3D observations with the temporally predicted actions. Such visual cues guide the diffusion model to minimize the following constraints:

$$\mathcal{L}_{refine} = L1(D, D^{gt}) + L1(\epsilon, \epsilon^{gt}) + BCE(g, g^{gt}) \tag{8}$$

where D represents denoising direction of gripper. ϵ is the noise added to the end-effector action. g is a binary variable that represents gripper's opening-closing action. $D^{gt}, \epsilon^{gt}, g^{gt}$ are their ground truth labels respectively. The denoised action sequence represents target end-effector poses in the world frame. These poses can be executed via inverse kinematics to reach the desired positions. Upon execution, the environment is updated and new observations are collected, enabling the next iteration of the control loop.

The entire control loop, comprising GAF's 4D scene representation, denoising framework and execution, is repeated iteratively until the manipulation task is completed. This closed-loop framework enables continuous adaptation to dynamic scene changes, leveraging GAF's spatiotemporal reasoning to maintain robust performance under occlusion and interaction uncertainties.

IV. Experiments

To validate the effectiveness of our approach, we perform extensive evaluations in both simulation and real-world environments. Furthermore, we conduct ablation studies, generalization tests, and multi-task experiments to thoroughly demonstrate the model's capabilities.

A. Comparison on Simulated Environments

Experiment Setup. We evaluate our method on RLBench[39] across 9 tasks, covering manipulation challenges including fine-grained placement, occlusion-rich interactions and articulated object handling. To ensure generalization, we randomly initialize the objects in the environment and collect 20 demonstrations of each individual task and tests across 100 different unseen poses for each task.

Baselines. For baseline selection, we compare our proposed ***V-4D-A*** paradigm against representative methods from the V-A and V-3D-A categories to highlight its advantages.

In the ***V-A*** category, we select Diffusion Policy (DP) [4] as a representative baseline, given its widespread adoption, open-source availability, and established performance in predicting

TABLE I: Success rates (%) comparison on RLBench.

Method	Toilet Seat Down	Open Grill	Close Grill	Close Fridge	Phone On Base
DP [4]	39	20	36	16	23
Act3D [7]	60	22	41	**47**	26
ManiGaussian [16]	34	24	38	41	30
Ours w/o GAF	57	16	49	27	31
Ours with GAF	**71**	**26**	**55**	42	**35**

Method	Lift Lid Up	Close Microwave	Push Button	Close Laptop	Avg.
DP [4]	81	62	61	64	44.7
Act3D [7]	91	73	60	58	53.1
ManiGaussian [16]	97	67	**63**	57	50.1
Ours w/o GAF	94	79	58	51	51.3
Ours with GAF	**100**	**85**	61	**69**	**60.4**

actions directly from 2D visual inputs. In the ***V-3D-A*** category, we use Act3D [7] as the baseline. It utilizes RGB-D inputs and leverages depth information along with camera parameters to lift 2D data into 3D space, generating a 3D scene feature cloud for action prediction.

We also include a comparison with ManiGaussian [16], which, like our method, is based on a Gaussian world model. Hyper-parameters such as prediction horizon, observation history or the number of trainable parameters were adjusted to match our method for fair comparison.

Results & Discussion. The experimental results in Table I demonstrate that our method outperforms all selected baselines. Compared to the V-A method Diffusion Policy (DP), our approach achieves a 15.7% improvement in success rate, particularly in tasks with heavy occlusion like "Toilet Seat Down", which highlights the critical role of 3D perception in robotic manipulation. Compared to the V-3D-A method Act3D, our method achieves a 7.3% improvement, indicating the effectiveness of incorporating future-aware 4D dynamic scene representations in manipulation tasks. Furthermore, when compared to ManiGaussian, our method yields a 10.3% increase in success rate, further validating the advantages of our explicit action generation and denoising strategy.

B. Comparison on Scene Reconstruction and Prediction

For scene reconstruction quality, we compare GAF against ManiGaussian [16], both of which generate Gaussian point clouds for current frame reconstruction and future frame prediction. We do not include dynamic reconstruction methods such as 4DGS [40] as baselines in this comparison, since their outputs are typically interpolations of the input images across both viewpoint and time. In other words, these methods can only reconstruct Gaussian corresponding to intermediate time points between input frames, rather than predicting future scene states from a given time point.

Qualitative Analysis. As shown in Fig. 4, our method achieves superior reconstruction fidelity and novel-view synthesis. ManiGaussian's renders (up) exhibit blurred textures and incomplete geometric details resulting in ambiguous

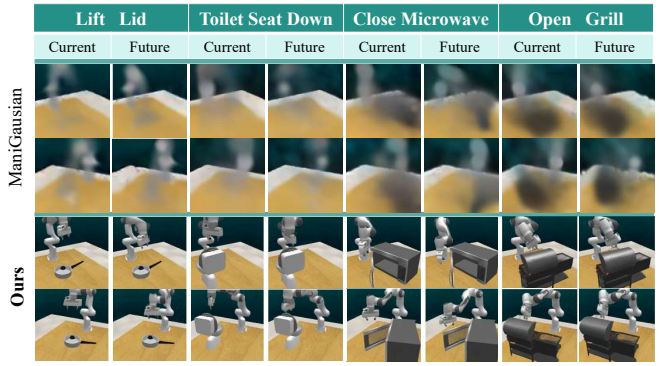

Fig. 4: Comparison of current scene reconstruction and future scene prediction from novel views.

spatial relationships. In contrast, our renders (down) preserve fine geometric structures, such as the gripper's articulated joints and object surfaces, even under partial observations. This clarity in reconstructing the Gaussian point cloud allows for the extraction of precise end-effector point clouds to calculate the action, which contributes to the fundamental difference compared to ManiGaussian.

Quantitative Metrics. We further evaluate reconstruction quality using standard metrics: PSNR (photometric fidelity), SSIM (structural similarity), and LPIPS (perceptual consistency). As shown in Table II, our method outperforms ManiGaussian by +11.5385 dB PSNR, +0.3864 SSIM, -0.5574 LPIPS on average across tasks in current scene reconstruction, and +10.5311 dB PSNR, +0.3856 SSIM, -0.5757 LPIPS in future state prediction. These metrics confirm that our dynamic rendering framework ensures high quality geometric accuracy and temporal coherence.

C. Ablation Study

Ablation on GAF. To evaluate the contribution of GAF, we remove this component and directly predict actions from input images using a diffusion model that denoises from random noise, without multi-view rendering or init action priors. With the ablation study results demonstrate in Table I, W/o GAF method drop our full method by -9.1% in average success rate across tasks. These results demonstrate that GAF's 4D representation improves action prediction accuracy.

Ablation on Denoising Framework. We set up a control group that directly executes the init actions. Experimental observations in Fig. 5 show that the arm fails to reach the correct position but continues pushing forward without action refinement. These results demonstrate that although GAF performs well in the early stages, reconstruction errors caused by partial occlusions lead to misaligned contacts during interaction, ultimately resulting in task failure. Therefore, action refinement is crucial for the successful completion of the entire task.

D. Spatial Generalization

Experiment Setup. To evaluate whether the model can achieve spatial generalization to object positions, we adopt a data collection strategy to ensure comprehensive spatial

TABLE II: Current & Future Novel view synthesis performance Comparison.

Method	Close Microwave			Toilet Seat Down			Lift Lid Up			**Average**		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
ManiGaussian[16] /Now	16.4274	0.3753	0.7628	16.5628	0.3976	0.6806	16.1492	0.4139	0.6217	16.3798	0.3956	0.6884
ManiGaussian[16] /Future	16.1368	0.3565	0.7896	15.7953	0.3687	0.7161	15.3727	0.3969	0.6572	15.7683	0.3740	0.7210
Ours / Now	**27.0986**	**0.7976**	**0.1291**	**28.1652**	**0.7779**	**0.1352**	**28.4912**	**0.7705**	**0.1286**	**27.9183**	**0.7820**	**0.1310**
Ours / Future	**24.5881**	**0.7650**	**0.1489**	**27.2951**	**0.7655**	**0.1456**	**27.0150**	**0.7483**	**0.1413**	**26.2994**	**0.7596**	**0.1453**

↑: Higher is better; ↓: Lower is better.

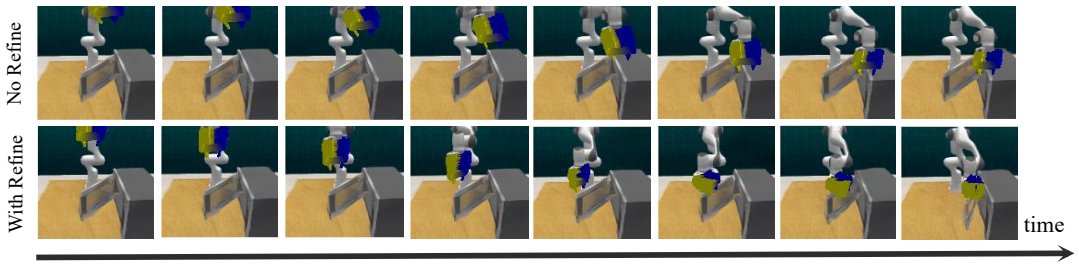

Fig. 5: **Ablation on Denoising Framework** The upper image shows a failed experiment without action denoising, while the lower image depicts a successful experiment after action denoising.

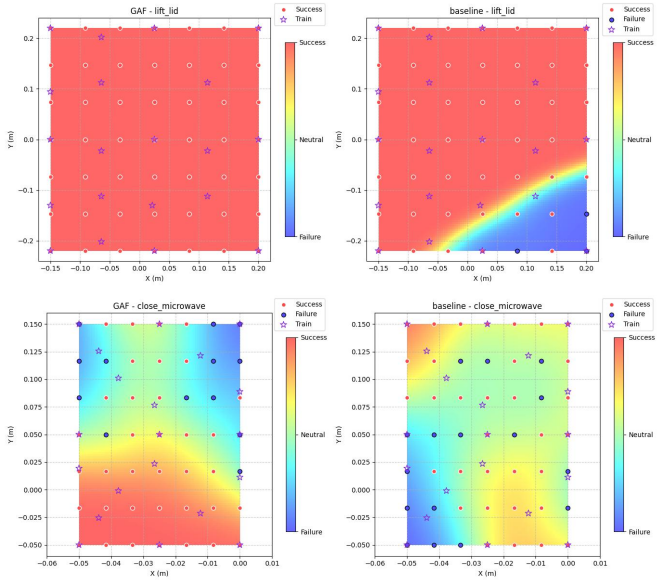

Fig. 6: **Spatial Generalization.** Outcome of our method and baseline trained on 20 demonstrations (purple stars). Red and blue colors representing successes and failures, respectively.

coverage of the operational workspace on 2 tasks on RLBench. During evaluation, we employ a grid sampling methodology across the entire workspace, which guarantees sufficient spatial variation and measurement consistency.

Results & Discussion. As illustrated in Fig. 6, the baseline DP encounters challenges when objects are placed along the boundaries and corners of the workspace. In contrast, our method achieves superior spatial generalization capability even when objects are placed on boundaries. Besides, our method is less sensitive to corners.

TABLE III: Success rates (%) with deltas of our method and baseline trained jointly on 4 tasks on RLBench.

Method	Toilet Seat Down	Close Microwave	LIFT LID Up	Close Laptop	**Average**
DP [4]	55 (+16)	50 (-12)	39 (-42)	18 (-46)	40.5 (-21)
Ours	**59 (-12)**	**85 (+0)**	**57 (-43)**	**79 (+10)**	**70 (-11.25)**

E. Multi-task Evaluation

Experiment Setup. In previous experiments, we trained the model for each task. To validate the generalization capabilities, we test its capacity to learn multiple tasks simultaneously. We train a single network using data collected from 4 RLBench tasks, 20 demonstrations each. Object positions are randomly initialized in both data collection and model evaluation phase.

Results & Discussion. As table III illustrated (The values in parentheses represent the performance change compared to single-task training), our method's average success rate only declines 11.25%. Our success rate exhibits the most significant decline in the "lift lid up" task, which is markedly distinct from the other three tasks. Nevertheless, in comparison to the substantial 21% decline observed in the baseline, our method demonstrates considerably superior performance. This shows our robust multi-tasking capabilities, demonstrating its effectiveness and potential as a world model.

F. Real-World Deployment

Experiment Setup. We evaluate our method using a real Franka Emika Panda robot equipped with Panda Hand on 5 tasks as shown in Fig. 8. We use 3 calibrated RealSense D435i cameras: two external camera and one mounted on

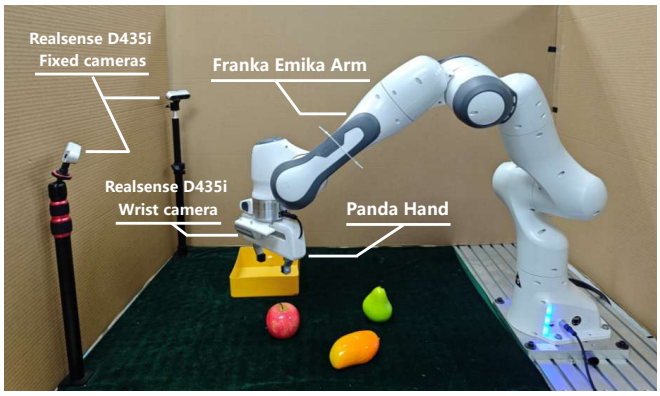

Fig. 7: **Real-World Experiment Setup** comprising a Franka arm with Panda hand, equipped with two static cameras and one wrist-mounted camera for visual input.

the wrist of the robot. During data collection, we record camera intrinsics and extrinsics. GAF uses only intrinsics for 4D reconstruction, while the extrinsics are employed in the subsequent action refinement stage to render gripper poses for visual alignment. For each task, we collect 20 demonstrations which tries to guarantee a good coverage of the workspace.

Results & Discussion. From Table IV, we can see that our method is capable of completing tasks in the real world, where different noise sources such as imperfect camera calibration are present. Failure cases were primarily observed in the Place Apple in a Box and Open Door task. Failures frequently occurred during the attempt to grasp the apple or door handle, likely due to the absence of force feedback. This limitation could potentially be addressed by augmenting the training data to better represent gripper state transitions or incorporating tactile sensing along with other sensors.

Push Button	10 / 10
Close Door	8 / 10
Open Door	7 / 10
Pick Cup	7 / 10
Place Apple	6 / 10

TABLE IV: Success Rate in real world tasks.

G. Implement Details

Training Phase. GAF is end-to-end trained using RGB video frames for supervision. We initialize the ViT with the weights from MASt3R [41], while the remaining layers are initialized randomly. For a fair comparison, all methods, including the baselines, use observations (128×128 in simulator and 1280×720 in the real world) from two external cameras and another wrist camera. It have been trained for 80k iterations with a batch size of 16 using a single NVIDIA RTX A800 GPU, taking approximately 24 hours.

In the action denoising process, we use 50 diffusion ierations based on DDIM [42]. To obtain more precise local observations, we incorporated the GT wrist camera data as an auxiliary resource. We use 1 last observations as input and predict 8 future actions. It have been trained for 50k iterations in 1.5 days on a single NVIDIA RTX A4090 GPU without extensive optimisation.

Evaluation Phase. Different from training, during inference we only perform 3 diffusion iterations, making our online deployment more real-time. Each 8-step action prediction takes less than 0.3 seconds. In real-world deployment, we further eliminate inference-induced delays by running action prediction and execution in parallel using separate threads.

V. Discussion

We present a 4D representation GAF to formulate a V-4D-A paradigm that infers future scene evolution from current visual observations to guide robotic manipulation. GAF supports scene reconstruction, future prediction, and action generation within a unified framework. This feed-forward pipeline requires only sparse-view RGB images and supports real-time execution. Experiments demonstrate that GAF achieves superior performance in both reconstruction quality and success rate, and can be successfully deployed in real-world environments. While our current method focuses on geometric modeling and motion prediction, it lacks semantic or task-level understanding. Future work will incorporate language modeling to bring high-level semantic priors into the system to support context-aware manipulation.

References

[1] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, P. Dollár, and R. Girshick, "Segment anything," 2023. [Online]. Available: https://arxiv.org/abs/2304.02643

[2] L. Yang, B. Kang, Z. Huang, X. Xu, J. Feng, and H. Zhao, "Depth anything: Unleashing the power of large-scale unlabeled data," in *CVPR*, 2024.

[3] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," 2021. [Online]. Available: https://arxiv.org/abs/2010.11929

[4] C. Chi, Z. Xu, S. Feng, E. Cousineau, Y. Du, B. Burchfiel, R. Tedrake, and S. Song, "Diffusion policy: Visuomotor policy learning via action diffusion," *The International Journal of Robotics Research*, p. 02783649241273668, 2023.

[5] M. Shridhar, Y. L. Lo, and S. James, "Generative image as action models," *arXiv preprint arXiv:2407.07875*, 2024.

[6] K. Black, M. Nakamoto, P. Atreya, H. R. Walke, C. Finn, A. Kumar, and S. Levine, "Zero-shot robotic manipulation with pretrained image-editing diffusion models," *ArXiv*, vol. abs/2310.10639, 2023. [Online]. Available: https://api.semanticscholar.org/CorpusID:264172455

[7] T. Gervet, Z. Xian, N. Gkanatsios, and K. Fragkiadaki, "Act3d: Infinite resolution action detection transformer for robotic manipulation," *arXiv preprint arXiv:2306.17817*, vol. 1, no. 3, 2023.

[8] Y. Ze, G. Zhang, K. Zhang, C. Hu, M. Wang, and H. Xu, "3d diffusion policy: Generalizable visuomotor policy learning via simple 3d representations," 2024. [Online]. Available: https://arxiv.org/abs/2403.03954

[9] Y. Ze, G. Yan, Y.-H. Wu, A. Macaluso, Y. Ge, J. Ye, N. Hansen, L. E. Li, and X. Wang, "Gnfactor: Multi-task real robot learning with generalizable neural feature fields," in *Conference on Robot Learning*. PMLR, 2023, pp. 284–301.

[10] C. Gao, Z. Xue, S. Deng, T. Liang, S. Yang, L. Shao, and H. Xu, "Riemann: Near real-time se (3)-equivariant robot manipulation without point cloud segmentation," *arXiv preprint arXiv:2403.19460*, 2024.

[11] S. Chen, R. G. Pinel, C. Schmid, and I. Laptev, "Polarnet: 3d point clouds for language-guided robotic manipulation," *ArXiv*, vol. abs/2309.15596, 2023. [Online]. Available: https://api.semanticscholar.org/CorpusID:263153114

[12] M. Shridhar, L. Manuelli, and D. Fox, "Perceiver-actor: A multi-task transformer for robotic manipulation," *ArXiv*, vol. abs/2209.05451, 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:252199474

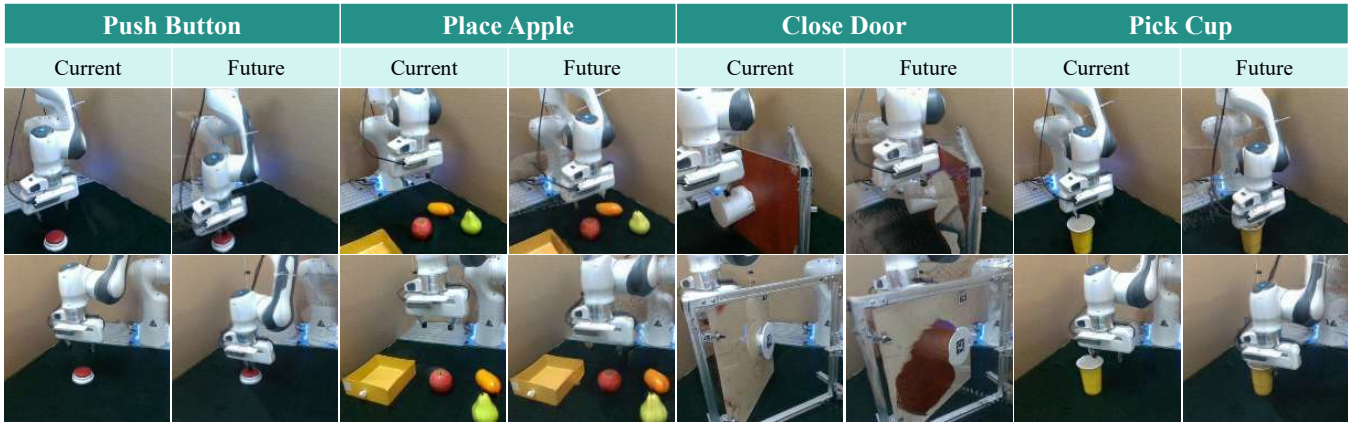

Fig. 8: **Real-World Experiment GAF Results** : rendered images from current Gaussian and future Gaussian

[13] I.-C. A. Liu, S. He, D. Seita, and G. Sukhatme, "Voxact-b: Voxel-based acting and stabilizing policy for bimanual manipulation," in *Conference on Robot Learning*, 2024. [Online]. Available: https://api.semanticscholar.org/CorpusID:271039193

[14] J. Zhang, Y. Chen, Y. Xu, Z. Huang, Y. Zhou, Y. Yuan, X. Cai, G. Huang, X. Quan, H. Xu, and L. Zhang, "4d-vla: Spatiotemporal vision-language-action pretraining with cross-scene calibration," *arXiv preprint arXiv:2506.22242*, 2025.

[15] D. Niu, Y. Sharma, H. Xue, G. Biamby, J. Zhang, Z. Ji, T. Darrell, and R. Herzig, "Pre-training auto-regressive robotic models with 4d representations," *arXiv preprint arXiv:2502.13142*, 2025.

[16] G. Lu, S. Zhang, Z. Wang, C. Liu, J. Lu, and Y. Tang, "Manigaussian: Dynamic gaussian splatting for multi-task robotic manipulation," *arXiv preprint arXiv:2403.08321*, 2024.

[17] G. Lu, B. Jia, P. Li, Y. Chen, Z. Wang, Y. Tang, and S. Huang, "Gwm: Towards scalable gaussian world models for robotic manipulation," *Proceedings of International Conference on Computer Vision (ICCV)*, 2025.

[18] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, "3d gaussian splatting for real-time radiance field rendering," 2023. [Online]. Available: https://arxiv.org/abs/2308.04079

[19] D. Ha and J. Schmidhuber, "Recurrent world models facilitate policy evolution," *Advances in neural information processing systems*, vol. 31, 2018.

[20] T. Chen, Y. Mu, Z. Liang, Z. Chen, S. Peng, Q. Chen, M. Xu, R. Hu, H. Zhang, X. Li, and P. Luo, "G3flow: Generative 3d semantic flow for pose-aware and generalizable object manipulation," *ArXiv*, vol. abs/2411.18369, 2024. [Online]. Available: https://api.semanticscholar.org/CorpusID:274305943

[21] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, X. Chen, K. Choromanski, T. Ding, D. Driess, A. Dubey, C. Finn, P. Florence, C. Fu, M. G. Arenas, K. Gopalakrishnan, K. Han, K. Hausman, A. Herzog, J. Hsu, B. Ichter, A. Irpan, N. Joshi, R. Julian, D. Kalashnikov, Y. Kuang, I. Leal, L. Lee, T.-W. E. Lee, S. Levine, Y. Lu, H. Michalewski, I. Mordatch, K. Pertsch, K. Rao, K. Reymann, M. Ryoo, G. Salazar, P. Sanketi, P. Sermanet, J. Singh, A. Singh, R. Soricut, H. Tran, V. Vanhoucke, Q. Vuong, A. Wahid, S. Welker, P. Wohlhart, J. Wu, F. Xia, T. Xiao, P. Xu, S. Xu, T. Yu, and B. Zitkovich, "Rt-2: Vision-language-action models transfer web knowledge to robotic control," 2023. [Online]. Available: https://arxiv.org/abs/2307.15818

[22] X. Chen, J. Guo, T. He, C. Zhang, P. Zhang, D. C. Yang, L. Zhao, and J. Bian, "Igor: Image-goal representations are the atomic control units for foundation models in embodied ai," *arXiv preprint arXiv:2411.00785*, 2024.

[23] J. Lu, D. Batra, D. Parikh, and S. Lee, "Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks," 2019. [Online]. Available: https://arxiv.org/abs/1908.02265

[24] A. Kloss, M. Bauza, J. Wu, J. B. Tenenbaum, A. Rodriguez, and J. Bohg, "Accurate vision-based manipulation through contact reasoning," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 6738–6744.

[25] A. Goyal, V. Blukis, J. Xu, Y. Guo, Y.-W. Chao, and D. Fox, "Rvt-2: Learning precise manipulation from few demonstrations," 2024. [Online]. Available: https://arxiv.org/abs/2406.08545

[26] Z. Gao, Y. Mu, C. Chen, J. Duan, P. Luo, Y. Lu, and S. E. Li, "Enhance sample efficiency and robustness of end-to-end urban autonomous driving via semantic masked world model," *IEEE Transactions on Intelligent Transportation Systems*, 2024.

[27] D. Hafner, T. Lillicrap, J. Ba, and M. Norouzi, "Dream to control: Learning behaviors by latent imagination," *arXiv preprint arXiv:1912.01603*, 2019.

[28] D. Hafner, J. Pasukonis, J. Ba, and T. Lillicrap, "Mastering diverse domains through world models," *arXiv preprint arXiv:2301.04104*, 2023.

[29] N. Hansen, H. Su, and X. Wang, "Td-mpc2: Scalable, robust world models for continuous control," *arXiv preprint arXiv:2310.16828*, 2023.

[30] Y. Seo, D. Hafner, H. Liu, F. Liu, S. James, K. Lee, and P. Abbeel, "Masked world models for visual control," in *Conference on Robot Learning*. PMLR, 2023, pp. 1332–1344.

[31] R. Mendonca, S. Bahl, and D. Pathak, "Structured world models from human videos," *arXiv preprint arXiv:2308.10901*, 2023.

[32] G. Lu, Z. Wang, C. Liu, J. Lu, and Y. Tang, "Thinkbot: Embodied instruction following with thought chain reasoning," *arXiv preprint arXiv:2312.07062*, 2023.

[33] H. Zhen, X. Qiu, P. Chen, J. Yang, X. Yan, Y. Du, Y. Hong, and C. Gan, "3d-vla: A 3d vision-language-action generative world model," *ArXiv*, vol. abs/2403.09631, 2024. [Online]. Available: https://api.semanticscholar.org/CorpusID:268385444

[34] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," in *ECCV*, 2020.

[35] R. Ranftl, A. Bochkovskiy, and V. Koltun, "Vision transformers for dense prediction," *ArXiv preprint*, 2021.

[36] B. Ye, S. Liu, H. Xu, L. Xueting, M. Pollefeys, M.-H. Yang, and P. Songyou, "No pose, no problem: Surprisingly simple 3d gaussian splats from sparse unposed images," *arXiv preprint arXiv:2410.24207*, 2024.

[37] A. V. Segal, D. Hähnel, and S. Thrun, "Generalized-icp," in *Robotics: Science and Systems*, 2009. [Online]. Available: https://api.semanticscholar.org/CorpusID:231748613

[38] V. Vosylius, Y. Seo, J. Uruç, and S. James, "Render and diffuse: Aligning image and action spaces for diffusion-based behaviour cloning," *arXiv preprint arXiv:2405.18196*, 2024.

[39] S. James, Z. Ma, D. R. Arrojo, and A. J. Davison, "Rlbench: The robot learning benchmark & learning environment," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 3019–3026, 2020.

[40] Z. Yang, H. Yang, Z. Pan, and L. Zhang, "Real-time photorealistic dynamic scene representation and rendering with 4d gaussian splatting," in *International Conference on Learning Representations (ICLR)*, 2024.

[41] V. Leroy, Y. Cabon, and J. Revaud, "Grounding image matching in 3d with mast3r," in *European Conference on Computer Vision*. Springer, 2024, pp. 71–91.

[42] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," *arXiv preprint arXiv:2010.02502*, 2020.