

UCLA

UCLA Electronic Theses and Dissertations

Title

Generalizable 3D Understanding and Generation

Permalink

<https://escholarship.org/uc/item/1m29k1nz>

Author

Xu, Chao

Publication Date

2025

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA

Los Angeles

Generalizable 3D Understanding and Generation

A dissertation submitted in partial satisfaction
of the requirements for the degree
Doctor of Philosophy in Statistics

by

Chao Xu

2025

© Copyright by
Chao Xu
2025

ABSTRACT OF THE DISSERTATION

Generalizable 3D Understanding and Generation

by

Chao Xu

Doctor of Philosophy in Statistics

University of California, Los Angeles, 2025

Professor Ying Nian Wu, Chair

How should we model the physical world for embodied agents? To carry out mundane tasks, humans exhibit remarkable capabilities in world modeling, mentally constructing 3D scenes from limited viewpoints, leveraging strong perceptual and generative priors to infer occluded regions and understand environmental affordances for interaction. Crucially, these abilities generalize across diverse scenes and objects. While modern foundation models display a similar breadth in language and in 2D domains, transferring such generalizability to 3D representations remains a key challenge for creating truly capable embodied agents.

This dissertation explores methods that enhance the generalization capacity of 3D vision systems for both understanding and generation. Key contributions include: devising efficient feed-forward pipelines that leverage 2D pre-trained priors to generate high-quality 3D shapes from challenging inputs such as a single image or sparse, unposed views; developing techniques to enhance the multi-view consistency and fidelity of generated 3D assets; and introducing methods for functional, part-level 3D understanding that discover cross-category affordances.

Collectively, these efforts significantly advance the state-of-the-art in robust, generalizable 3D generation from limited views and in achieving more fine-grained, functional object understanding. This dissertation systematically rethinks and refines the pipelines to unite the strengths of 2D generative priors with 3D representation learning. The

research presented provides foundational steps towards more scalable, generalizable, and functionally aware 3D vision systems. The dissertation concludes by summarizing these contributions and outlining promising directions for future research in pursuit of physically grounded artificial intelligence.

The dissertation of Chao Xu is approved.

Hao Su

Hongjing Lu

Demetri Terzopoulos

Ying Nian Wu, Committee Chair

University of California, Los Angeles

2025

*To my parents —
whose unwavering love and support light the way.*

TABLE OF CONTENTS

1	Introduction	1
2	2D Priors for Efficient 3D Generation	5
2.1	Related Work	8
2.1.1	3D Generation Guided by 2D Prior Models	8
2.1.2	Single Image to 3D	9
2.1.3	Generalizable Neural Reconstruction	10
2.2	Method: One-2-3-45	11
2.2.1	Zero123: View-Conditioned 2D Diffusion	12
2.2.2	Can NeRF Optimization Lift Multi-View Predictions to 3D?	13
2.2.3	Neural Surface Reconstruction from Imperfect Multi-View Predictions	14
2.2.4	Camera Pose Estimation	16
2.3	Experiments	17
2.3.1	Implementation Details	17
2.3.2	Single Image to 3D Mesh	18
2.3.3	Ablation Study	21
2.3.4	Text to 3D Mesh	23
2.4	More Examples on Real-World Images	23
2.5	Failure Cases and Limitations	23
2.6	Discussion	25
3	Multi-View Consistency and Fidelity in 3D Generation	27
3.1	Related Work	28
3.1.1	3D Generation	28

3.1.2	Sparse View Reconstruction	30
3.2	Method: One-2-3-45++	31
3.2.1	Consistent Multi-View Generation	32
3.2.2	3D Diffusion with Multi-View Condition	34
3.2.3	Texture Refinement	38
3.3	Experiments	38
3.3.1	Comparison on Image to 3D	38
3.3.2	Comparison on Text to 3D	40
3.3.3	Analyses	42
3.4	Discussion	44
3.5	Related Project	45
3.5.1	MeshFormer	45
4	Sparse-View 3D Reconstruction and Pose Estimation	47
4.1	Related Work	50
4.1.1	Sparse-View 3D Reconstruction	50
4.1.2	Pose-Free Reconstruction	51
4.1.3	Open-World 3D Generation	52
4.2	Method: SpaRP	53
4.2.1	Tiling Sparse View Images as Input Condition	54
4.2.2	Image-to-NOCS Diffusion as a Pose Estimator	56
4.2.3	Multi-View Prediction for 3D Reconstruction	58
4.2.4	Pose Refinement with Reconstructed 3D Model	59
4.3	Experiments	60
4.3.1	Evaluation Settings	60

4.3.2	Experiment Results	63
4.3.3	Analysis	65
4.4	Additional Real-World Examples	68
4.5	Discussion	68
5	Generalizable Part-Level 3D Understanding	69
5.1	Related Work	72
5.2	Task Definition	74
5.3	Method: PartAfford	75
5.3.1	Abstraction Encoder	75
5.3.2	Affordance Decoder	76
5.4	Part Affordance Dataset	78
5.5	Experiments	80
5.5.1	Experimental Settings	81
5.5.2	Implementation Details	83
5.5.3	Results and Analysis	83
5.5.4	Model Generalization	84
5.5.5	Failure Cases	85
5.6	Further Analysis	86
5.7	Discussion	89
5.8	Related Project	89
5.8.1	GAPartNet	89
6	Conclusion	91
	References	93

LIST OF FIGURES

2.1	One-2-3-45	6
2.2	Pipeline of One-2-3-45	11
2.3	Traditional NeRF optimization cannot lift multi-view predictions to 3D . . .	13
2.4	Analysis of the novel view synthesis quality of Zero123	14
2.5	Qualitative results: One-2-3-45 on synthetic and real images	17
2.6	Qualitative comparison: One-2-3-45 vs. single-image-to-3D baselines	18
2.7	One-2-3-45 avoids the multi-face (Janus) problem	19
2.8	Error distribution of predicted elevations	19
2.9	Ablation study: Training strategies and the number of input views	20
2.10	Qualitative comparison: 360° reconstruction vs. multi-view fusion	22
2.11	Incorrect input elevations lead to distorted reconstruction	22
2.12	Qualitative comparison: One-2-3-45 vs. text-to-3D baselines	23
2.13	Qualitative comparison: One-2-3-45 vs. Shap-E on real-world images	24
2.14	Failure cases of One-2-3-45	25
3.1	One-2-3-45++	28
3.2	Pipeline of One-2-3-45++	29
3.3	Consistent multi-view generation	32
3.4	Multi-view local condition	36
3.5	Qualitative results: One-2-3-45++ vs. single-image-to-3D baselines	37
3.6	User study of One-2-3-45++	39
3.7	Qualitative results: One-2-3-45++ on synthetic images	40
3.8	Qualitative comparison: One-2-3-45++ vs. text-to-3D baselines	41
3.9	Pipeline of MeshFormer	45

4.1	SpaRP	49
4.2	Pipeline of SpaRP	54
4.3	Input and output of the multi-view diffusion model in SpaRP	55
4.4	Qualitative comparison: SpaRP vs. image(s)-to-3D baselines	62
4.5	Single-view vs. sparse-view for 3D reconstruction	64
4.6	Ablation study: Pose refinement	65
4.7	Qualitative results: SpaRP on real-world images	67
5.1	PartAfford	70
5.2	Pipeline of PartAfford	74
5.3	Part affordance dataset	79
5.4	Qualitative results: Predicting affordances across object categories	80
5.5	Experiment: PartAfford vs. affordance discovery baselines and ablations	82
5.6	PartAfford can generalize to unseen object categories	86
5.7	Failure cases of PartAfford	86
5.8	Visualization of the iterative affordance discovery mechanism	86
5.9	Ablation study: Number of slots	87
5.10	Overview of GAPartNet	90

LIST OF TABLES

2.1	Quantitative comparison: One-2-3-45 vs. single-image-to-3D baselines	19
3.1	Quantitative comparison: One-2-3-45++ vs. single-image-to-3D baselines . .	38
3.2	Quantitative comparison: One-2-3-45++ vs. text-to-3D baselines	41
3.3	Ablation study: Effect of proposed modules in One-2-3-45++	42
3.4	Ablation study: Effect of 3D diffusion module design in One-2-3-45++ . . .	43
3.5	Quantitative comparison: One-2-3-45++ vs. multi-view generation baselines	44
4.1	Quantitative comparison: SpaRP vs. camera pose estimation baselines . . .	61
4.2	Quantitative comparison: SpaRP vs. image(s)-to-3D baselines	63
4.3	Ablation study: Effect of the number of input views in SpaRP	64
4.4	Ablation study: Effect of the number of experts in SpaRP	66
4.5	Ablation study: Effect of joint training in SpaRP	66
5.1	Experiment: Predicting sittable affordance	84
5.2	Experiment: Predicting other affordances	85

ACKNOWLEDGMENTS

First and foremost, I owe my deepest gratitude to my advisors—Dr. Ying Nian Wu, Dr. Song-Chun Zhu, and Dr. Hao Su. Their guidance has shaped not only this dissertation but also my way of thinking as a researcher. Dr. Wu’s blend of sharp insight and disciplined rigor continually challenged and inspired me, both in my research endeavors and personal growth. He has an artist’s touch for breaking down complex knowledge and concepts, making them accessible and engaging. Dr. Zhu presented a unified blueprint for AI that truly opened my door to the field of general artificial intelligence research and set a benchmark for intellectual taste. To Dr. Su, I owe a debt of gratitude for teaching me the crucial importance of seeking out fundamental questions, courageously challenging the unknown, and always anchoring research in solid principles. His visionary guidance and genuine support consistently propelled my work forward.

I would also like to extend my heartfelt thanks to my other two committee members, Dr. Hongjing Lu and Dr. Demetri Terzopoulos, for their thoughtful feedback and continual encouragement.

My PhD journey was made immeasurably richer by the wonderful opportunity to collaborate with many brilliant co-authors (listed alphabetically): Ang Li, Chengyang Zhao, Chong Zeng, Haian Jin, Hansheng Chen, Haoran Geng, He Wang, Helin Xu, Hongzhi Wu, Isabella Liu, Jiayuan Gu, Li Yi, Linghao Chen, Mengqi Zhang, Minghua Liu, Mukund Varma T, Ruoxi Shi, Siyuan Huang, Xiaoshuai Zhang, Xinyue Wei, Yixin Chen, Yixin Zhu, Yulin Liu, Zexiang Xu, Zhaoning Wang, and Zhuoyang Zhang. Working with you made my research not only more productive but also profoundly more enjoyable and inspiring.

A special shout-out to Minghua, Haian, and Linghao: Building a pioneering system that held its own against OpenAI’s concurrent release on day one was both nerve-racking and electrifying, and I am incredibly grateful we pushed through the pain together. Every late-night Zoom sprint and spirited brainstorm with you sharpened my thinking and truly brought our ambitious vision to reality. I learned immensely from each of you.

In addition, I sincerely thank all my labmates at the VCLA lab at UCLA and the Su Lab at UC San Diego. The vibrant discussions, shared camaraderie, and supportive environment you all fostered have been precious. A special thank you to Yixin Zhu, my steadfast mentor and “big-brother.” Your guidance, advocacy, and friendship have been invaluable to both my research and my career.

I am also deeply grateful to my mentor Albert Pumarola and manager Ali Thabet, my peer Yuming Du, and all other Metamates in the GenAI media foundation model team. Working with you on cutting-edge, industry-scale models at Meta was transformative—it significantly broadened my horizons and enriched my understanding of building and shipping efficient real-world AI applications.

My gratitude further extends to all my friends, both within the research community and in everyday life. I will always cherish the conference-hall coffee chats, Friday-night foodie quests, and refreshing mountain hikes we shared. Those moments of connection and respite were a vital source of inspiration.

I also want to express my appreciation to all the artists, developers, and 3D enthusiasts who tried our models through demos and code, filed issues, and shared awe-inspiring creations. Seeing your excitement is the most rewarding proof that our work matters beyond the pages of this dissertation.

Lastly, and most importantly, I want to express my deepest gratitude to my parents, Lijuan and Jianhuan. You nurtured my curiosity, embraced my quirks, and listened without judgment. Your quiet sacrifices and unwavering faith turned a first-generation student into someone unafraid of improbable goals. I am, and always will be, indebted to you beyond words.

VITA

- 2016–2020 B.Eng. in Electronics and Computer Engineering, Zhejiang University.
- 2016–2020 B.S. in Computer Engineering, University of Illinois Urbana-Champaign.
- 2024–2025 Research Scientist Intern, Hillbot Inc.
- 2024–2025 Research Scientist Intern, Meta.
- 2020–2025 Graduate Student Researcher & Teaching Fellow in Statistics, University of California, Los Angeles.

PUBLICATIONS

MeshFormer: High-Quality Mesh Generation with 3D-Guided Reconstruction Model. M. Liu*, C. Zeng*, X. Wei, R. Shi, L. Chen, **C. Xu**, M. Zhang, Z. Wang, X. Zhang, I. Liu, H. Wu, H. Su. *NeurIPS*, 2024.

SpaRP: Fast 3D Object Reconstruction and Pose Estimation from Sparse Views. **C. Xu**, A. Li, L. Chen, Y. Liu, R. Shi, H. Su[†], M. Liu[†]. *ECCV*, 2024.

One-2-3-45++: Fast Single Image to 3D Objects with Consistent Multi-View Generation and 3D Diffusion. M. Liu*, R. Shi*, L. Chen*, Z. Zhang*, **C. Xu***, X. Wei, H. Chen, C. Zeng, H. Su. *CVPR*, 2024.

One-2-3-45: Any Single Image to 3D Mesh in 45 Seconds without Per-Shape Optimization. M. Liu*, **C. Xu***, H. Jin*, L. Chen*, M. V. T, Z. Xu, H. Su. *NeurIPS*, 2023.

GAPartNet: Cross-Category Domain-Generalizable Object Perception and Manipulation via Generalizable and Actionable Parts. H. Geng*, H. Xu*, C. Zhao*, **C. Xu**, L. Yi, S. Huang, H. Wang. *CVPR*, 2023.

* denotes equal contribution. [†] denotes equal advisory.

PartAfford: Part-level Affordance Discovery from Cross-category 3D Objects. **C. Xu**, Y. Chen, H. Wang, S.-C. Zhu, Y. Zhu, S. Huang. *ECCVW*, 2022.

CHAPTER 1

Introduction

Humans navigate and interact with the three-dimensional physical world with a fluency that often belies its underlying complexity. We grasp a mug without triangulating its pose, infer the hidden rear of a vehicle while merging onto a freeway, and mentally align a bookcase with an empty corner before tightening a single screw. Each act presupposes an internal, geometry-aware model of physical reality—rich enough to predict contact, robust to occlusion, and fast enough to support real-time action.

Recognizing this fluency, we argue that the next decisive frontier for artificial general intelligence (AGI) is fundamentally physical: agents must be able to perceive, imagine, and act in 3D space. Cognitive science lends empirical support. The classic mental-rotation experiments (Shepard and Metzler, 1971) demonstrated that humans mentally spin shapes at a speed proportional to the rotation angle, revealing an analog spatial code in the mind. At the computational level, Marr (1982) argued that vision’s task extends beyond image processing to construct representations making explicit the world’s shape and spatial organization, culminating in a 3D model suitable for downstream cognition and control. These observations converge on a simple claim: An embodied agent needs a spatial substrate that mediates sensation and action.

Over the recent years, computer vision has pursued that substrate through successively richer 3D representations. Explicit meshes and point clouds first surfaced with ShapeNet (Chang et al., 2015); implicit surfaces, signed-distance fields (SDFs), and neural radiance fields (NeRFs) soon enhanced the fidelity and detail of per-scene reconstructions. Today, a single neural field can capture sub-millimeter details in complex scenes. Yet expressiveness is not generalizability. A network trained extensively on datasets like

ShapeNet may recognize numerous chairs but often fails to generalize to dissimilar categories, such as a luggage trolley, let alone generate a plausible 3D model of one from a single photograph. Open-world competence—the hallmark of modern language and 2D vision models—remains elusive in 3D. Without generalization, 3D vision may carefully interpolate among chairs in a room yet miss the elephant—capturing the physical world’s diversity demands more than per-scene precision.

We identify three intertwined factors that impede the development of broadly generalizable 3D models. First, *data scale*: Current 3D datasets, even large ones like Objaverse containing millions of models, are orders of magnitude smaller than the multi-billion-pair image-text datasets fueling 2D generative models, and data capturing physical interactions remain scarcer still. Second, *distribution breadth*: Previous 3D datasets often exhibit biases towards common categories like furniture and vehicles, frequently comprising synthetic CAD assets, leaving a long tail of object categories under-represented. Third, *semantic granularity*: Many approaches treat geometry monolithically, neglecting the functionally salient parts—handles, surfaces, articulated components—that afford interaction and define an object’s utility. Without tackling these factors, learned models acquire impressive scene-specific or category-limited fidelity yet falter when asked to extrapolate across viewpoints, categories, or functional tasks.

Addressing the generalization dilemma in 3D necessitates efforts in both 3D generation and 3D understanding. Richard Feynman noted, “What I cannot create, I do not understand.” This aphorism highlights a fundamental duality between generation and understanding. To understand an object well enough to grasp it, a system must predict its hidden geometry; to generate a plausible shape from sparse observations, it must respect structural and functional constraints.

This dissertation follows this philosophy to tackle generalizable 3D generation and understanding.

Concretely, for open-world 3D generation tasks, we unite the strengths of two existing experts: 2D *generalists* and 3D *specialists*. Pre-trained 2D generative models, scaled

on massive internet data, act as powerful *generalists* capable of synthesizing diverse visual concepts, yet their inherent understanding often remains confined to the 2D plane. Conversely, many established 3D algorithms excel as *specialists*, achieving high fidelity from dense, well-posed views but struggling with ambiguity or sparse inputs. We challenge both: Tasking 2D priors to generate 3D-consistent information suitable for reconstruction, while pushing 3D methods to handle the inconsistencies and sparsity inherent in generated or real-world sparse views. This strategy, bridging the expansive knowledge of the 2D *generalists* with the geometric precision of the 3D *specialists*, motivates our contributions in One-2-3-45, One-2-3-45++, SpaRP, and MeshFormer. These works progressively develop methods for efficient, high-quality 3D object generation from single or sparse images, overcoming limitations of prior optimization-heavy or view-inconsistent approaches.

Complementing generation, we address generalizable 3D understanding by focusing on semantic granularity, specifically at the part level. We argue that true object understanding requires recognizing not just the object’s category but its functional components. From a task-oriented perspective, this involves identifying object parts and inferring their affordances (what interactions they afford) and actionability (how they can be manipulated) in a category-agnostic manner. We develop algorithms capable of discovering functionally relevant parts directly from local geometry across object categories. This line of inquiry leads to the development of PartAfford and GAPartNet, focusing on discovering part-level affordances and understanding articulated part actionability across diverse object types, using novel datasets annotated specifically for these tasks.

By weaving generation and understanding into a single learning loop, we aim to transform 3D representations from static reconstructions into mental workspaces capable of supporting perception, imagination, and downstream action.

The chapters that follow instantiate this agenda through new algorithms, benchmarks, and large-scale experiments, charting a course toward 3D models that not only see but also create—and, by Feynman’s criterion, truly understand—the physical world.

Chapter 2 proposes One-2-3-45, establishing a novel feed-forward paradigm for single-

image-to-3D generation. This work leverages view-conditioned 2D diffusion models to synthesize multi-view images, which are then efficiently reconstructed into a 3D mesh by a generalizable neural surface reconstruction module. This approach significantly accelerates the generation process compared to prior optimization-based methods and bridges large-scale 2D priors with explicit 3D representations.

Chapter 3 presents One-2-3-45++, which addresses the key limitation of multi-view inconsistency inherent in models like Zero123 used in the original One-2-3-45. It introduces techniques for generating more 3D-consistent multi-view images jointly and employs a robust multi-view conditioned 3D diffusion model for reconstruction, markedly improving output quality and fidelity. This chapter also includes a pointer to Meshformer, a related project focusing on high-quality mesh generation using 3D-native guidance.

Chapter 4 details SpaRP, extending the generation capabilities from a single image to unposed, sparse-view inputs. This work tackles the joint challenge of estimating relative camera poses and reconstructing a 3D object when input views lack overlap and pose information. It utilizes a finetuned sparse-view conditioned diffusion model to predict both Normalized Object Coordinate Space (NOCS) maps for pose estimation and consistent multi-view images for reconstruction, demonstrating synergy between understanding input view relationships and generating the complete object.

Chapter 5 shifts focus to part-level 3D understanding with PartAfford. This chapter introduces the task of discovering part affordances directly from 3D geometry using only weak, object-level affordance set labels. We propose a method combining slot attention for part abstraction with affordance set matching and primitive regularization, and introduce a tailored part affordance dataset. A related project, GAPartNet, focusing on category-agnostic understanding of articulated part actionability, is briefly discussed.

Finally, Chapter 6 provides a summary of the key ideas and contributions explored throughout this dissertation. It reflects on the synergistic relationship between 3D generation and understanding for achieving generalization and discusses promising avenues for future research in creating more capable physically-grounded AI systems.

CHAPTER 2

2D Priors for Efficient 3D Generation

Single image 3D reconstruction, the task of reconstructing a 3D model of an object from a single 2D image, is a long-standing problem in the computer vision community and is crucial for a wide range of applications, such as robotic object manipulation and navigation, 3D content creation, as well as AR/VR (Mees et al., 2019; Chiu et al., 2009; Yang et al., 2021). The problem is challenging as it requires not only the reconstruction of visible parts but also the hallucination of invisible regions. Consequently, this problem is often ill-posed and corresponds to multiple plausible solutions because of insufficient evidence from a single image. On the other hand, humans can adeptly infer unseen 3D content based on our extensive knowledge of the 3D world. To endow intelligent agents with this ability, many existing methods (Kanazawa et al., 2018; Girdhar et al., 2016; Huang et al., 2022; Choy et al., 2016; Xie et al., 2020; Yagubbayli et al., 2021; Fan et al., 2017; Wen et al., 2019; Liu et al., 2021; Chou et al., 2023; Liu et al., 2020) exploit class-specific priors by training 3D generative networks on 3D shape datasets (Chang et al., 2015). However, these methods often fail to generalize to unseen categories, and their reconstruction quality is constrained by the limited size of public 3D datasets.

In this chapter, we pursue a generic solution to turn an image of any object, regardless of its category, into a high-quality 3D textured mesh. To achieve this, we propose a novel approach that can effectively utilize the strong priors learned by 2D diffusion models for 3D reconstruction. Compared to 3D data, 2D images are more readily available and scalable. Recent 2D generative models (*e.g.*, DALL-E (Ramesh et al., 2021, 2022), Imagen (Saharia et al., 2022), and Stable Diffusion (Rombach et al., 2022)) and visual-language models (*e.g.*, CLIP (Radford et al., 2021)) have made significant strides by pre-training on Internet-scale



Figure 2.1: One-2-3-45 reconstructs a full 360° mesh of any object in 45 seconds given a single image of it. In each example, we showcase the input image in the left column, alongside the generated textured and textureless meshes from three different views.

image datasets. Since they learn a wide range of visual concepts and possess strong priors about our 3D world, it is natural to marry 3D tasks with them. Consequently, an emerging body of research (Jain et al., 2022; Hong et al., 2022; Michel et al., 2022; Poole et al., 2022; Lin et al., 2022), as exemplified by DreamField (Jain et al., 2022), DreamFusion (Poole et al., 2022), and Magic3D (Lin et al., 2022), employs 2D diffusion models or vision language models to assist 3D generative tasks. The common paradigm of them is to perform per-shape optimization with differentiable rendering and the guidance of the CLIP model or 2D diffusion models. While many other 3D representations have been explored, neural fields are the most commonly used representation during optimization.

Although these optimization-based methods have achieved impressive results on both text-to-3D (Poole et al., 2022; Jain et al., 2022; Lin et al., 2022) and image-to-3D

tasks (Melas-Kyriazi et al., 2023a; Seo et al., 2023), they face some common dilemmas: (a) **time-consuming**. Per-shape optimization typically involves tens of thousands of iterations of full-image volume rendering and prior model inferences, resulting in typically tens of minutes per shape. (b) **memory intensive**. Since the full image is required for the 2D prior model, the volume rendering can be memory-intensive when the image resolution goes up. (c) **3D inconsistent**. Since the 2D prior model only sees a single view at each iteration and tries to make every view look like the input, they often generate 3D inconsistent shapes (*e.g.*, with two faces, or the Janus problem (Melas-Kyriazi et al., 2023a; Poole et al., 2022)). (d) **poor geometry**. Many methods utilize the density field as the representation in volume rendering. It is common that they produce good RGB renderings but extracting high-quality mesh tends to be difficult.

In this work, instead of following the common optimization-based paradigm, we propose a novel approach to utilize 2D prior models for 3D modeling. At the heart of our approach is the combination of a 2D diffusion model with a cost-volume-based 3D reconstruction technique, enabling the reconstruction of a high-quality 360° textured mesh from a single image in a feed-forward pass without per-scene optimization. Specifically, we leverage a recent 2D diffusion model, Zero123 (Liu et al., 2023d), which is fine-tuned on Stable Diffusion (Rombach et al., 2022) to predict novel views of the input image given the camera transformation. We utilize it to generate multi-view predictions of the input single image so that we can leverage multi-view 3D reconstruction techniques to obtain a 3D mesh. There are two challenges associated with reconstruction from synthesized multi-view predictions: (a) the inherent lack of perfect consistency within the multi-view predictions, which can lead to severe failures in optimization-based methods such as NeRF methods (Mildenhall et al., 2021; Chen et al., 2022). (b) the camera pose of the input image is required but unknown. To tackle them, we build our reconstruction module upon a cost volume-based neural surface reconstruction approach, SparseNeuS (Long et al., 2022), which is a variant of MVSNerF (Chen et al., 2021). Additionally, we introduce a series of essential training strategies that enable the reconstruction of 360-degree meshes from inherently inconsistent multi-view predictions. We also propose an elevation estimation

module that estimates the elevation of the input shape in Zero123’s canonical coordinate system, which is used to compute the camera poses required by the reconstruction module.

By integrating the three modules of multi-view synthesis, elevation estimation, and 3D reconstruction, our method can reconstruct 3D meshes of any object from a single image in a feed-forward manner. Without costly optimizations, our method reconstructs 3D shapes in significantly less time, *e.g.*, in just 45 seconds. Our method favors better geometry due to the use of SDF representations, and generates more consistent 3D meshes, thanks to the camera-conditioned multi-view predictions. Moreover, our reconstruction adheres more closely to the input image compared to existing methods. See Figure 2.1 for some of our example results. We evaluate our method on both synthetic data and real images and demonstrate that our method outperforms existing methods in terms of both quality and efficiency.

2.1 Related Work

2.1.1 3D Generation Guided by 2D Prior Models

Recently, 2D generative models (*e.g.*, DALL-E (Ramesh et al., 2021, 2022), Imagen (Saharia et al., 2022), and Stable Diffusion (Rombach et al., 2022)) and vision-language models (*e.g.*, CLIP (Radford et al., 2021)) have learned a wide range of visual concepts by pre-training on Internet-scale image datasets. They possess powerful priors about our 3D world and have inspired a growing body of research to employ 2D prior models for assisting 3D understanding (Liu et al., 2023a,c) and generative tasks. Exemplified by DreamField (Jain et al., 2022), DreamFusion (Poole et al., 2022), and Magic3D (Lin et al., 2022), a line of works follows the paradigm of per-shape optimization. They typically optimize a 3D representation (*i.e.*, NeRF, mesh, SMPL human model) and utilize differentiable rendering to generate 2D images from various views. The images are then fed to the CLIP model (Hong et al., 2022; Jain et al., 2022; Michel et al., 2022; Lee and Chang, 2022; Canfes et al., 2023; Khalid et al., 2022; Aneja et al., 2022; Jetchev,

2021; Xu et al., 2023b; Liu et al., 2022b) or 2D diffusion model (Poole et al., 2022; Lin et al., 2022; Seo et al., 2023; Melas-Kyriazi et al., 2023a; Deng et al., 2022; Wang et al., 2022a; Xu et al., 2023a; Metzger et al., 2023; Zhou and Tulsiani, 2023; Raj et al., 2023; Tang et al., 2023c) for calculating the loss functions, which are used to guide the 3D shape optimization. In addition to optimization-based 3D shape generation, some works train a 3D generative model but leverage the embedding space of CLIP (Cheng et al., 2022; Liu et al., 2023f; Sanghi et al., 2022), and some works focus on generating textures or materials for input meshes using 2D models’ prior (Michel et al., 2022; Wei et al., 2023; Chen et al., 2023a; Metzger et al., 2023; Richardson et al., 2023).

2.1.2 Single Image to 3D

Before the emergence of CLIP and large-scale 2D diffusion models, people often learn 3D priors from 3D synthetic data (Chang et al., 2015) or real scans (Reizenstein et al., 2021). Unlike 2D images, 3D data can be represented in various formats and numerous representation-specific 3D generative models have been proposed. By combining 2D image encoder and 3D generators, they generate 3D data in various representations, including 3D voxels (Girdhar et al., 2016; Wu et al., 2017; Choy et al., 2016; Xie et al., 2020, 2019; Yagubbayli et al., 2021), point clouds (Fan et al., 2017; Yang et al., 2018; Groueix et al., 2018; Achlioptas et al., 2018; Melas-Kyriazi et al., 2023b; Zeng et al., 2022), polygon meshes (Kanazawa et al., 2018; Wang et al., 2018; Wen et al., 2019; Nash et al., 2020), and parametric models (Pavlakos et al., 2019; Zuffi et al., 2018, 2017). Recently, there has been an increasing number of work on learning to generate a 3D implicit field from a single image (Xu et al., 2019; Mescheder et al., 2019; Saito et al., 2019; Huang et al., 2022; Park et al., 2019; Gao et al., 2022; Gupta et al., 2023; Jang and Agapito, 2021; Müller et al., 2022; Wu et al., 2023b; Mittal et al., 2022).

As previously mentioned, several recent works leverage 2D diffusion models to perform per-shape optimization, allowing for the text-to-3D task (Poole et al., 2022; Lin et al., 2022; Jain et al., 2022) given that diffusion models are typically conditioned on text. To

enable the generation of 3D models from a single image, some works (Melas-Kyriazi et al., 2023a; Deng et al., 2022; Metzger et al., 2023) utilize textual inversion (Gal et al., 2022), to find the best-matching text embedding for the input image, which is then fed into a diffusion model. NeuralLift-360 (Hu et al., 2021) adds a CLIP loss to enforce similarity between the rendered image and the input image. 3DFuse (Seo et al., 2023) finetunes the Stable Diffusion model with LoRA layers (Hu et al., 2021) and a sparse depth injector to ensure greater 3D consistency. A recent work Zero123 (Liu et al., 2023d; Shi et al., 2023a) finetunes the Stable Diffusion model (Saharia et al., 2022) to generate a novel view of the input image based on relative camera pose. In addition to these methods, OpenAI trains a 3D native diffusion model Point-E (Nichol et al., 2022), which uses several million internal 3D models to generate point clouds. Very recently, they published another model Shap-E (Jun and Nichol, 2023) which is trained to generate parameters of implicit functions that can be used for producing textured meshes or neural radiance fields.

2.1.3 Generalizable Neural Reconstruction

Traditional NeRF-like methods (Mildenhall et al., 2021; Wang et al., 2021a) use a neural network to represent a single scene and require per-scene optimization. However, some approaches aim to learn priors across scenes and generalize to novel scenes. These methods typically take a few source views as input and leverage 2D networks for extracting 2D features. The pixel features are then unprojected into 3D space, and a NeRF-based rendering pipeline is applied on top of them. In this way, they can generate a 3D implicit field given a few source views in a single feed-forward pass. Among the methods, some (Wang et al., 2021b; Reizenstein et al., 2021; Henzler et al., 2021; Yu et al., 2021a; Yang et al., 2023; Liu et al., 2022a; Kulhánek et al., 2022; Trevithick and Yang, 2021; Varma et al., 2022) directly aggregate 2D features with MLPs or transformers, while others explicitly construct the 3D feature/cost volume (Chen et al., 2021; Johari et al., 2022; Zhang et al., 2022b; Long et al., 2022), and utilize the voxel feature for decoding density and color. In addition to the density field representation, some methods such as

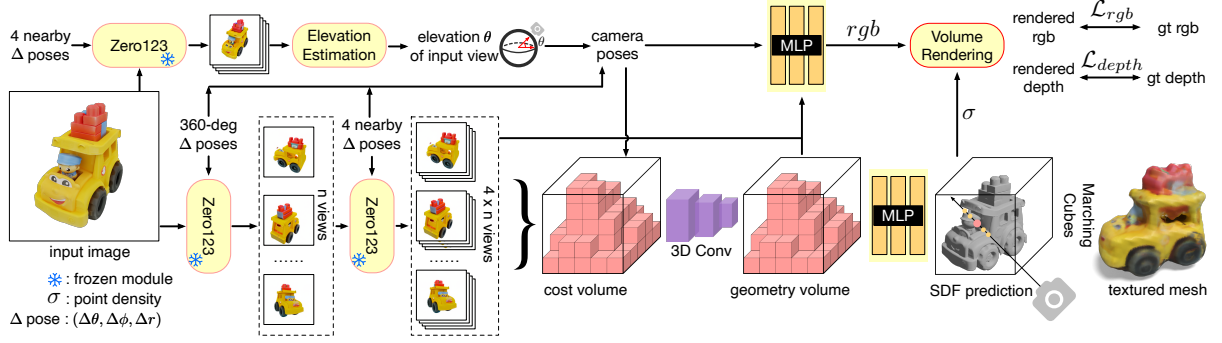


Figure 2.2: Our method consists of three primary components: (a) **Multi-view synthesis**: we use a view-conditioned 2D diffusion model, Zero123 (Liu et al., 2023d), to generate multi-view images in a two-stage manner. The input of Zero123 includes a single image and a relative camera transformation, which is parameterized by the relative spherical coordinates $(\Delta\theta, \Delta\phi, \Delta r)$. (b) **Pose estimation**: we estimate the elevation angle θ of the input image based on four nearby views generated by Zero123. We then obtain the poses of the multi-view images by combining the specified relative poses with the estimated pose of the input view. (c) **3D reconstruction**: We feed the multi-view posed images to an SDF-based generalizable neural surface reconstruction module for 360° mesh reconstruction.

SparseNeuS (Long et al., 2022) and VolRecon (Ren et al., 2022) utilize SDF representations for geometry reconstruction.

2.2 Method: One-2-3-45

Our overall pipeline is illustrated in Figure 2.2. In Section 2.2.1, we introduce a view-conditioned 2D diffusion model, Zero123 (Liu et al., 2023d), which is used to generate multi-view images. In Section 2.2.2, we show that traditional NeRF-based and SDF-based methods fail to reconstruct high-quality meshes from inconsistent multi-view predictions even given ground truth camera poses. Therefore, in Section 2.2.3, we propose a cost volume-based neural surface reconstruction module that can be trained to handle inconsistent multi-view predictions and reconstruct a 3D mesh in a single feed-forward pass. Specifically, we build upon the SparseNeuS (Long et al., 2022) and introduce several critical training strategies to support 360° mesh reconstruction. Additionally, in Section 2.2.4, we demonstrate the necessity of estimating the pose of the input view in Zero123’s canonical space for 3D reconstruction. While the azimuth and radius can be

arbitrarily specified, we propose a novel module that utilizes four nearby views generated by Zero123 to estimate the elevation of the input view.

2.2.1 Zero123: View-Conditioned 2D Diffusion

Recent 2D diffusion models (Ramesh et al., 2021; Saharia et al., 2022; Rombach et al., 2022) have demonstrated the ability to learn a wide range of visual concepts and strong priors by training on internet-scale data. While the original diffusion models mainly focused on the task of text-to-image, recent work (Zhang and Agrawala, 2023; Hu et al., 2021) has shown that fine-tuning pretrained models allows us to add various conditional controls to the diffusion models and generate images based on specific conditions. Several conditions, such as canny edges, user scribbles, depth, and normal maps, have already proven effective (Zhang and Agrawala, 2023).

The recent work Zero123 (Liu et al., 2023d) shares a similar spirit and aims to add view-point condition control for the Stable Diffusion model (Rombach et al., 2022). Specifically, given a single RGB image of an object and a relative camera transformation, Zero123 aims to control the diffusion model to synthesize a new image under this transformed camera view. To achieve this, Zero123 fine-tunes the Stable Diffusion on paired images with their relative camera transformations, synthesized from a large-scale 3D dataset (Deitke et al., 2022). During the creation of the fine-tuning dataset, Zero123 assumes that the object is centered at the origin of the coordinate system and uses a spherical camera, *i.e.*, the camera is placed on the sphere’s surface and always looks at the origin. For two camera poses (θ_1, ϕ_1, r_1) and (θ_2, ϕ_2, r_2) , where θ_i , ϕ_i , and r_i denote the polar angle, azimuth angle, and radius, their relative camera transformation is parameterized as $(\theta_2 - \theta_1, \phi_2 - \phi_1, r_2 - r_1)$. They aim to learn a model f , such that $f(x_1, \theta_2 - \theta_1, \phi_2 - \phi_1, r_2 - r_1)$ is perceptually similar to x_2 , where x_1 and x_2 are two images of an object captured from different views. Zero123 finds that such fine-tuning enables the Stable Diffusion model to learn a generic mechanism for controlling the camera viewpoints, which extrapolates outside of the objects seen in the fine-tuning dataset.

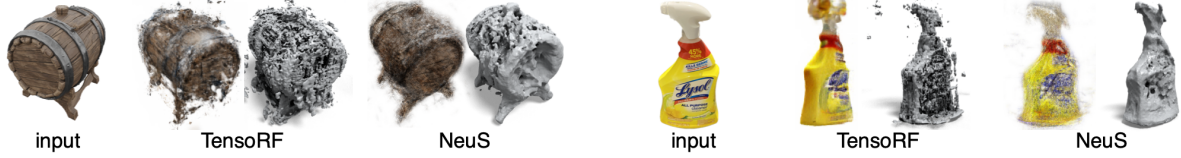


Figure 2.3: NeRF-based method (Mildenhall et al., 2021) and SDF-based method (Wang et al., 2021a) fail to reconstruct high-quality meshes given multi-view images predicted by Zero123. See Figure 2.1 for our reconstruction results.

2.2.2 Can NeRF Optimization Lift Multi-View Predictions to 3D?

Given a single image of an object, we can utilize Zero123 (Liu et al., 2023d) to generate multi-view images, but can we use traditional NeRF-based or SDF-based methods (Chen et al., 2022; Wang et al., 2021a) to reconstruct high-quality 3D meshes from these predictions? We conduct a small experiment to test this hypothesis. Given a single image, we first generate 32 multi-view images using Zero123, with camera poses uniformly sampled from the sphere surface. We then feed the predictions to a NeRF-based method (TensorRF (Mildenhall et al., 2021)) and an SDF-based method (NeuS (Wang et al., 2021a)), which optimize density and SDF fields, respectively. However, as shown in Figure 2.3, both methods fail to produce satisfactory results, generating numerous distortions and floaters. This is primarily due to the inconsistency of Zero123’s predictions. In Figure 2.4, we compare Zero123’s predictions with ground-truth renderings. We can see that the overall PSNR is not very high, particularly when the input relative pose is large or the target pose is at unusual locations (*e.g.*, from the bottom or the top). However, the mask IoU (most regions are greater than 0.95) and CLIP similarity are relatively good. This suggests that Zero123 tends to generate predictions that are perceptually similar to the ground truth and have similar contours or boundaries, but the pixel-level appearance may not be exactly the same. Nevertheless, such inconsistencies between the source views are already fatal to traditional optimization-based methods. Although the original Zero123 paper proposes another method for lifting its multi-view predictions, we will demonstrate in experiments that it also fails to yield perfect results and entails time-consuming optimization.

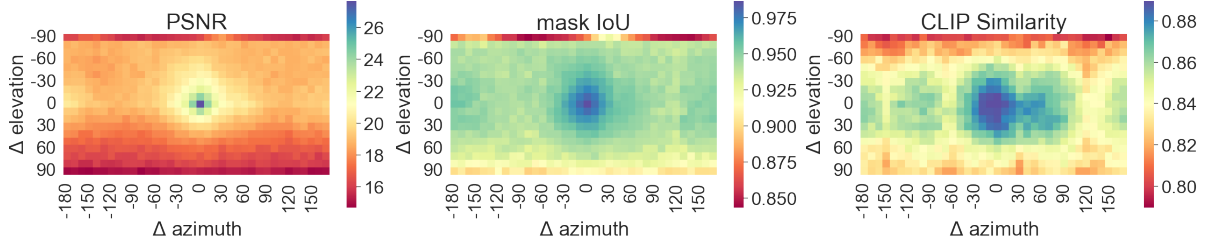


Figure 2.4: We analyze the prediction quality of Zero123 by comparing its predictions to ground truth renderings across various view transformations. For each view transformation, we report the average PSNR, mask IoU, and CLIP similarity of 100 shapes from the Objaverse (Deitke et al., 2022) dataset. The prediction mask is calculated by considering foreground objects (*i.e.*, non-white regions). Zero123 provides more accurate predictions when the view transformation is small.

2.2.3 Neural Surface Reconstruction from Imperfect Multi-View Predictions

Instead of using optimization-based approaches, we base our reconstruction module on a generalizable SDF reconstruction method SparseNeuS (Long et al., 2022), which is essentially a variant of the MVNeRF (Chen et al., 2021) pipeline that combines multi-view stereo, neural scene representation, and volume rendering. As illustrated in Figure 2.2, our reconstruction module takes multiple source images with corresponding camera poses as input and generates a textured mesh in a single feed-forward pass. In this section, we will first briefly describe the network pipeline of the module and then explain how we train the module, select the source images, and generate textured meshes. Additionally, in Section 2.2.4, we will discuss how we generate the camera poses for the source images.

As shown in Figure 2.2, our reconstruction module takes m posed source images as input. The module begins by extracting m 2D feature maps using a 2D feature network. Next, the module builds a 3D cost volume whose contents are computed by first projecting each 3D voxel to m 2D feature planes and then fetching the variance of the features across the m projected 2D locations. The cost volume is then processed using a sparse 3D CNN to obtain a geometry volume that encodes the underlying geometry of the input shape. To predict the SDF at an arbitrary 3D point, an MLP network takes the 3D coordinate and its corresponding interpolated features from the geometry encoding volume as input.

To predict the color of a 3D point, another MLP network takes as input the 2D features at the projected locations, interpolated features from the geometry volume, and the viewing direction of the query ray relative to the viewing direction of the source images. The network predicts the blending weights for each source view, and the color of the 3D point is predicted as the weighted sum of its projected colors. Finally, an SDF-based rendering technique is applied on top of the two MLP networks for RGB and mask rendering (Wang et al., 2021a). In each iteration, we randomly choose one view to build the cost volume and another view for rendering supervision.

2-Stage Source View Selection and Groundtruth-Prediction Mixed Training.

Although the original SparseNeuS (Long et al., 2022) paper only demonstrated frontal view reconstruction, we have extended it to reconstruct 360-degree meshes in a single feed-forward pass by selecting source views in a particular way. Specifically, our reconstruction model is trained on a 3D object dataset while freezing Zero123. We follow Zero123 to normalize the training shapes and use a spherical camera model. For each shape, we first render n ground-truth RGB images from n camera poses uniformly placed on the sphere. For each of the n views, we use Zero123 to predict four nearby views. During training, we feed all $4 \times n$ predictions with ground-truth poses into the reconstruction module and randomly choose one of the n ground-truth RGB images views as the target view. We call this view selection strategy as *2-stage source view selection*. We supervise the training with both the ground-truth RGB and mask values. In this way, the module can learn to handle the inconsistent predictions from Zero123 and reconstruct a consistent 360° mesh. We argue that our two-stage source view selection strategy is critical since uniformly choosing $n \times 4$ source views from the sphere surface would result in larger distances between the camera poses. However, cost volume-based methods (Long et al., 2022; Johari et al., 2022; Chen et al., 2021) typically rely on very close source views to find local correspondences. Furthermore, as shown in Figure 2.4, when the relative pose is small (*e.g.*, 10 degrees apart), Zero123 can provide very accurate and consistent predictions and thus can be used to find local correspondences and infer the geometry.

During training, we utilize n ground-truth renderings in the initial stage. We find that

employing n predicted images at this stage would suffer from notable inconsistencies across different views, complicating the network’s ability to learn sharp details (see examples in ablation study). However, during inference, we can replace the n ground-truth renderings with Zero123 predictions, as shown in Figure 2.2, the network can automatically generalize to some extent. We will show in the experiments that this groundtruth-prediction mixed training strategy is also important. To export the textured mesh, we use marching cubes (Lorensen and Cline, 1987) to extract the mesh from the predicted SDF field and query the color of the mesh vertices as described in (Wang et al., 2021a). Although our reconstruction module is trained on a 3D dataset, we find that it mainly relies on local correspondences and can generalize to unseen shapes very well.

2.2.4 Camera Pose Estimation

Our reconstruction module requires camera poses for the $4 \times n$ source view images. Note that we adopt Zero123 for image synthesis, which parameterizes cameras in a canonical spherical coordinate frame, (θ, ϕ, r) , where θ , ϕ and r represent the elevation, azimuth, and radius. While we can arbitrarily adjust the azimuth angle ϕ and the radius r of all source view images simultaneously, resulting in the rotation and scaling of the reconstructed object accordingly, this parameterization requires knowing the absolute elevation angle θ of one camera to determine the relative poses of all cameras in a standard XYZ frame. More specifically, the relative poses between camera (θ_0, ϕ_0, r_0) and camera $(\theta_0 + \Delta\theta, \phi_0 + \Delta\phi, r_0)$ vary for different θ_0 even when $\Delta\theta$ and $\Delta\phi$ are the same. Because of this, changing the elevation angles of all source images together (*e.g.*, by 30 degrees up or 30 degrees down) will lead to the distortion of the reconstructed shape (see Figure 2.11 for examples).

Therefore, we propose an elevation estimation module to infer the elevation angle of the input image. First, we use Zero123 to predict four nearby views of the input image. Then we enumerate all possible elevation angles in a coarse-to-fine manner. For each elevation candidate angle, we compute the corresponding camera poses for the four images and calculate a reprojection error for this set of camera poses to measure the



Figure 2.5: Qualitative examples of One-2-3-45 for both synthetic and real images. Each triplet showcases an input image, a textured mesh, and a textureless mesh.

consistency between the images and the camera poses. The elevation angle with the smallest reprojection error is used to generate the camera poses for all $4 \times n$ source views by combining the pose of the input view and the relative poses.

2.3 Experiments

2.3.1 Implementation Details

For each input image, we generate $n = 8$ images by choosing camera poses uniformly placed on the sphere surface and then generate 4 local images (10° apart) for each of the 8 views, resulting in 32 source-view images for reconstruction. During training, we freeze the Zero123 (Liu et al., 2023d) model and train our reconstruction module on the Objaverse-LVIS (Deitke et al., 2022) dataset, which contains 46K 3D models in 1,156 categories. We use BlenderProc (Denninger et al., 2023) to render ground-truth RGB images. For images with background, we utilize an off-the-shelf segmentation network SAM (Kirillov et al., 2023) with bounding-box prompts for background removal.



Figure 2.6: We compare One-2-3-45 with Point-E (Nichol et al., 2022), Shap-E (Jun and Nichol, 2023), Zero123 (Stable Dreamfusion version) (Liu et al., 2023d), 3DFuse (Seo et al., 2023), and RealFusion (Melas-Kyriazi et al., 2023a). In each example, we present both the textured and textureless meshes. As 3DFuse (Seo et al., 2023) and RealFusion (Melas-Kyriazi et al., 2023a) do not natively support the export of textured meshes, we showcase the results of volume rendering instead.

2.3.2 Single Image to 3D Mesh

We present qualitative examples of our method in Figures 2.1 and 2.5, illustrating its effectiveness in handling both synthetic images and real images. We also compare One-2-3-45 with existing zero-shot single image 3D reconstruction approaches, including Point-E (Nichol et al., 2022), Shap-E (Jun and Nichol, 2023), Zero123 (Stable Dreamfusion version) (Liu et al., 2023d), 3DFuse (Seo et al., 2023), and RealFusion (Melas-Kyriazi et al., 2023a). Among them, Point-E and Shap-E are two 3D native diffusion models released by OpenAI, which are trained on several million internal 3D data, while others are optimization-based approaches leveraging priors from Stable Diffusion (Rombach

Table 2.1: Quantitative Comparison on GSO (Downs et al., 2022) and Objaverse (Deitke et al., 2022) datasets.

	Prior Source	F-Score			CLIP Similarity			Time
		GSO	Obj.	avg.	GSO	Obj.	avg.	
Point-E (Nichol et al., 2022)	internal	81.0	81.0	81.0	74.3	78.5	76.4	78s
Shap-E (Jun and Nichol, 2023)	3D data	<u>83.4</u>	<u>81.2</u>	<u>82.3</u>	79.6	82.1	80.9	27s
Zero123+SD (Liu et al., 2023d)	2D diffusion models	75.1	69.9	72.5	71.0	72.7	71.9	~15min
RealFusion (Melas-Kyriazi et al., 2023a)		66.7	59.3	63.0	69.3	69.5	69.4	~90min
3DFuse (Seo et al., 2023)		60.7	60.2	60.4	71.4	74.0	72.7	~30min
Ours		84.0	83.1	83.5	<u>76.4</u>	<u>79.7</u>	<u>78.1</u>	45s



Figure 2.7: One-2-3-45 generates 3D consistent objects, resolving the Janus problem prevalent in prior works.

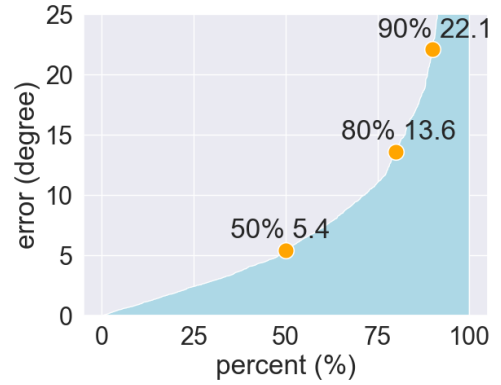


Figure 2.8: Error distribution of predicted elevations. The median and average are 5.4 and 9.7 degrees.

et al., 2022).

Figure 2.6 presents the qualitative comparison. While most methods can generate plausible 3D meshes from a single image, notable differences exist among them in terms of geometry quality, adherence to the input, and overall 3D consistency. In terms of geometry quality, approaches like RealFusion (Melas-Kyriazi et al., 2023a) and 3DFuse (Seo et al., 2023), which optimize a neural radiance field, face challenges in extracting high-quality meshes. Likewise, Point-E (Nichol et al., 2022) produces a sparse point cloud as its output, resulting in numerous holes on the reconstructed meshes. In contrast, our approach utilizes an SDF presentation and favors better geometry. Regarding adherence to the input, we observe that most baseline methods struggle to preserve the similarity to the input image. Although Shap-E performs slightly better, it still produces lots of failure

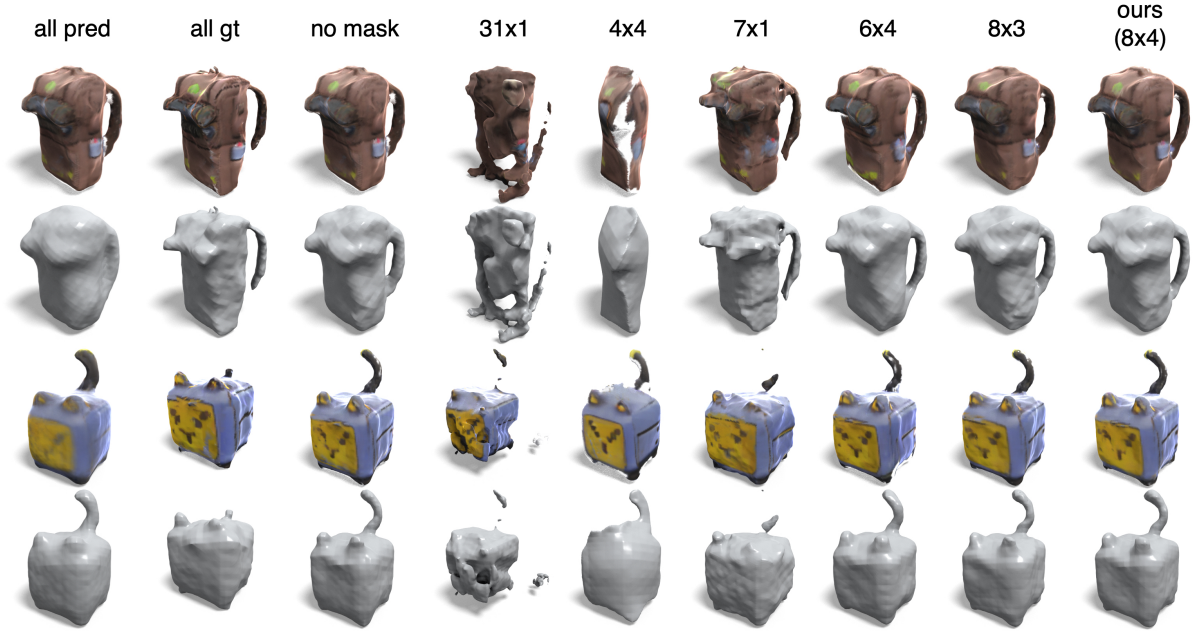


Figure 2.9: Ablations on training strategies of the reconstruction module and the number of views.

cases (see the backpack without shoulder straps, distorted shoe, and stool with three legs). In contrast, our approach leverages a powerful 2D diffusion model to directly produce high-quality multi-view images, rather than relying on 3D space hallucination. This strategy provides better adherence to the input views, alleviates the burden of the 3D reconstruction module, and yields results that are more finely attuned to the input. Furthermore, many approaches encounter challenges in achieving consistent 3D results (also known as the Janus problem (Melas-Kyriazi et al., 2023a; Poole et al., 2022)), as highlighted in Figure 2.7 (two-handle mug, multi-face Mario, and two-face backpack). One of the contributing factors to this issue is that several methods optimize each view independently, striving to make each view resemble the input. In contrast, our method capitalizes on the view-conditioned 2D diffusion model, inherently enhancing 3D consistency.

We also quantitatively compare the approaches on Objaverse (Deitke et al., 2022) and GoogleScannedObjects (GSO) (Downs et al., 2022) datasets. For each dataset, we randomly choose 20 shapes and render a single image per shape for evaluation. To align the predictions with the ground-truth mesh, we linearly search the scaling factor and

the rotation angle, apply Iterative Closest Point (ICP) for sampled point clouds, and select the one with the most number of inliers. We follow RealFusion (Melas-Kyriazi et al., 2023a) to report F-score (with a threshold of 0.05) and CLIP similarity, and the runtime on an A100 GPU. As shown in Table 2.1, our method outperforms all baseline approaches in terms of F-Score. As for CLIP similarity, we surpass all methods except a concurrent work Shap-E (Jun and Nichol, 2023). We find that CLIP similarity is very sensitive to the color distribution and less discriminative in local geometry variations (*i.e.*, the number of legs of a stool, the number of handles of a mug). Regarding running time, our method demonstrates a notable advantage over optimization-based approaches and performs on par with 3D native diffusion models, such as Point-E (Nichol et al., 2022) and Shap-E (Jun and Nichol, 2023). Specifically, our 3D reconstruction module reconstructs a 3D mesh in approximately 5 seconds, with the remaining time primarily spent on Zero123 predictions, which takes roughly 1 second per image on an A100 GPU.

2.3.3 Ablation Study

Training strategies. We ablate our training strategies in Figure 2.9. We found that without our 2-stage source view selection strategy, a network trained to consume 31 uniformly posed Zero123 predictions (fourth column) suffers from severe inconsistency among source views, causing the reconstruction module to fail completely. If we feed only 7 source views (sixth column) without the four nearby views, the reconstruction fails to capture local correspondence and cannot reconstruct fine-grained geometry. During training, we first render n ground-truth renderings and then use Zero123 to predict four nearby views for each of them. If we train directly on 8×4 ground-truth renderings without Zero123 prediction during training (second column), it fails to generalize well to Zero123 predictions during inference, with many missing regions. Instead, if we replace the n ground-truth renderings with n Zero123 predictions during training (first column), the network fail to generate sharp details (see the strips of the backpack).

Elevation estimation. Our reconstruction module relies on accurate elevation angles

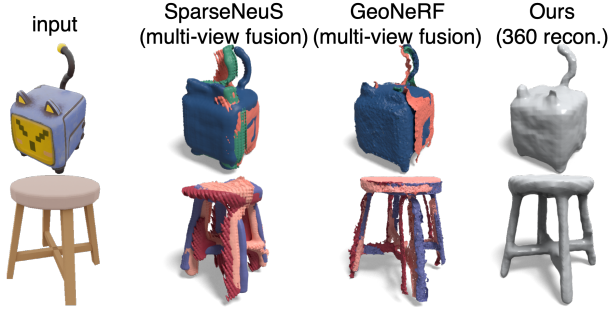


Figure 2.10: 360° reconstruction vs. multi-view fusion. Meshes from different views are in different colors.



Figure 2.11: Incorrect elevations lead to distorted reconstruction. Our elevation estimation module can predict an accurate elevation of the input view.

of the input view. In Figure 2.11, we demonstrate the impact of providing incorrect elevation angles (*e.g.*, altering the elevation angles of source views by $\pm 30^\circ$), which results in distorted reconstruction results. Instead, utilizing our predicted elevation angles can perfectly match results with ground truth elevations. We also quantitatively test our elevation estimation module by rendering 1,700 images from random camera poses. As shown in Figure 2.8, our elevation estimation module predicts accurate elevations.

Number of source views. In Figure 2.9, we also investigate the impact of varying the number of source views on 3D reconstruction. We observe that our method is not very sensitive to the number of views as long as the reconstruction module is retrained with the corresponding setting.

360° reconstruction vs. multi-view fusion. While our method reconstructs a 360° mesh in a single pass, most existing generalizable neural reconstruction approaches (Long et al., 2022; Johari et al., 2022; Chen et al., 2021) primarily focus on frontal view reconstruction. An alternative approach is to independently infer the geometry for each view and subsequently fuse them together. However, we have observed that this strategy often struggles with multi-view fusion due to inconsistent Zero123 predictions, as illustrated in Figure 2.10.

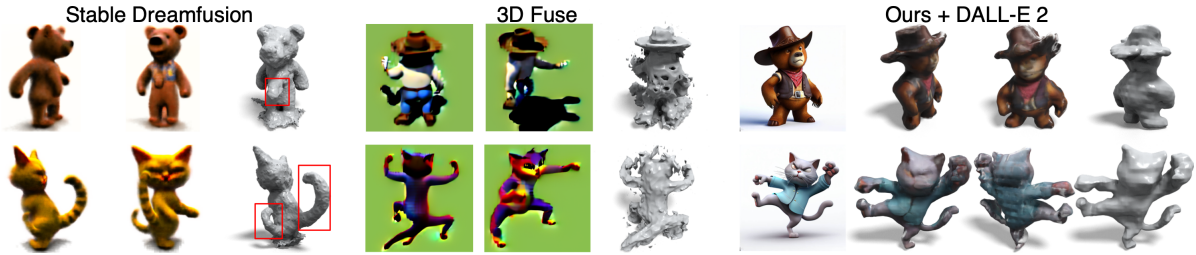


Figure 2.12: Text to 3D. First row: “a bear in cowboy suit.” Second row: “a kungfu cat.” We utilize DALL-E 2 (Ramesh et al., 2022) to generate an image conditioned on the text and then lift it to 3D. We compare our method with Stable Dreamfusion (Poole et al., 2022) and 3DFuse (Seo et al., 2023). For baselines, volume renderings are shown.

2.3.4 Text to 3D Mesh

As shown in Figure 2.12, by integrating with off-the-shelf text-to-image 2D diffusion models (Rombach et al., 2022; Ramesh et al., 2022), our method can be naturally extended to support text-to-image-3D tasks and generate high-quality textured meshes in a short time. See supplementary for more examples.

2.4 More Examples on Real-World Images

In Figure 2.13, we showcase more examples on real-world images and compare our method with the concurrent method Shap-E (Jun and Nichol, 2023). The input images are from unsplash.com or captured by ourselves. Note that our results exhibit a closer adherence to the input image.

2.5 Failure Cases and Limitations

Our method relies on Zero123 for generating multi-view images, which introduces challenges due to its occasional production of inconsistent results. In Figure 2.14, we present two typical cases that exemplify such inconsistencies. The first case involves an input view that lacks sufficient information, such as the back view of a fox. In this scenario, Zero123 struggles to generate consistent predictions for the invisible regions, such as the face of the fox. As a consequence, our method may encounter difficulties in accurately

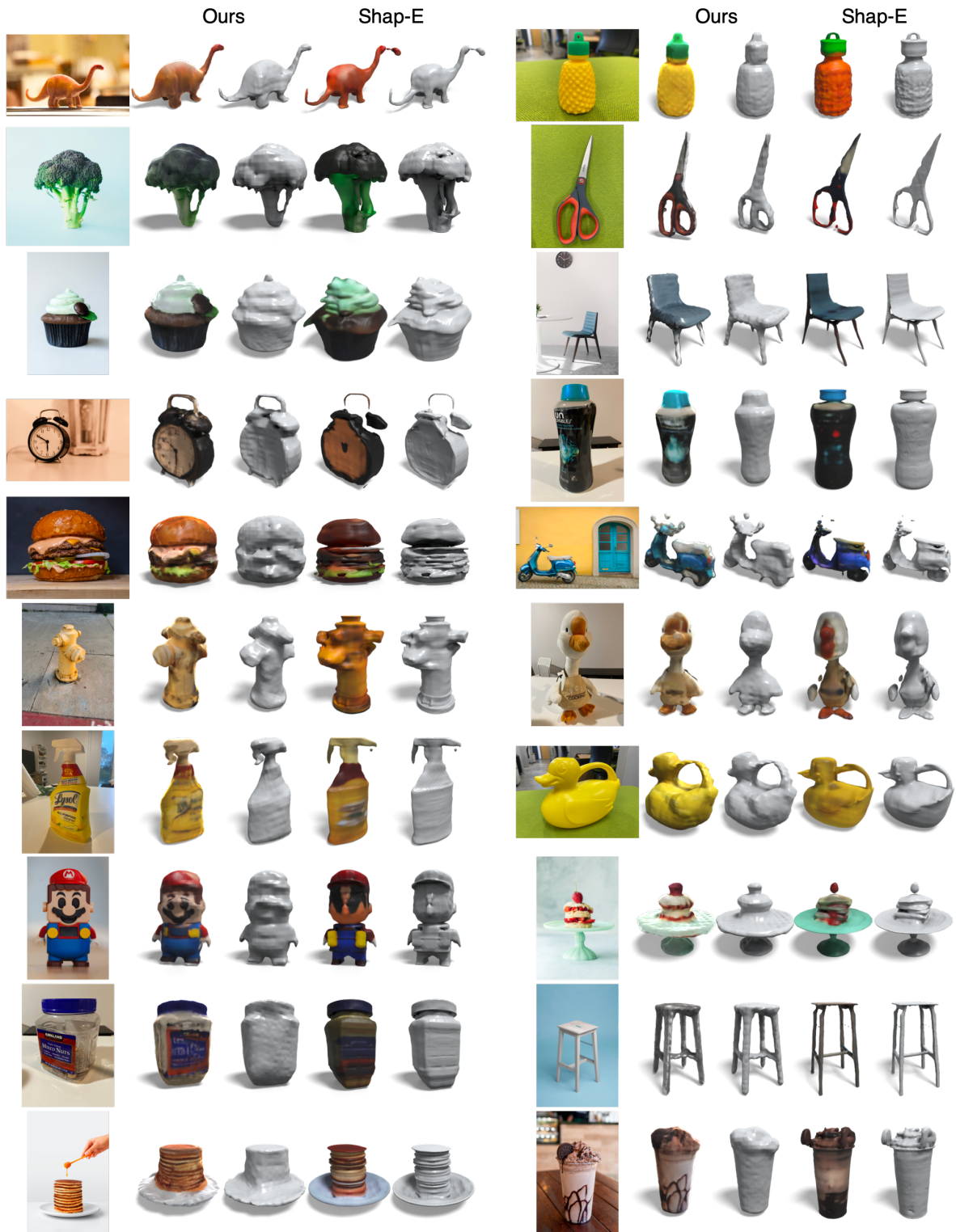


Figure 2.13: We compare One-2-3-45 with Shap-E (Jun and Nichol, 2023) on real-world images. In each example, we present the input image, generated textured and textureless meshes.

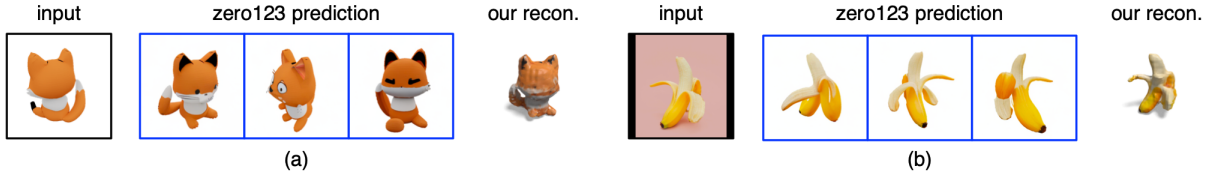


Figure 2.14: Failure cases. Our method relies on Zero123 to generate multi-view images, and we encounter challenges when Zero123 generates inconsistent results. (a) The input view lacks sufficient information. (b) The input view contains ambiguous or complicated structures.

inferring the geometry for those regions. The second case involves an input view with ambiguous or complex structures, such as the pulp and peel of a banana. In such situations, Zero123’s ability to accurately infer the underlying geometry becomes limited. As a result, our method may be affected by the inconsistent predictions generated by Zero123. It is important to acknowledge that these limitations arise from the occasional scenarios, and they can impact the performance of our method in certain cases. Addressing these challenges and refining the reliability of Zero123’s predictions remain areas for further investigation and improvement.

We have also noticed slight artifacts on the back side of our generated results. As one of the first works in combining view-conditioned 2D diffusion models with generalizable multi-view reconstruction, we believe that there is still ample room for exploring more advanced reconstruction techniques and incorporating additional regularizations. By doing so, we expect to significantly mitigate the minor artifacts and further enhance results in the future.

2.6 Discussion

In this chapter, we present One-2-3-45, a novel paradigm for reconstructing a high-quality 360° mesh of any object from a single image of it. By integrating view-conditioned 2D diffusion priors with a generalizable neural surface reconstruction module, we established a feed-forward pipeline that bypasses the need for time-consuming per-shape optimization. Compared to contemporary zero-shot methods, our approach demonstrates significant

advantages in geometric fidelity, 3D consistency, and faithfulness to the input view, while dramatically reducing processing time to under a minute. Furthermore, our method also allows for straightforward extension to text-to-3D generation.

While One-2-3-45 represents a significant step towards efficient and generalizable 3D content creation, its reliance on the multi-view synthesis capabilities of the underlying 2D diffusion model remains a critical factor. The quality and, crucially, the 3D consistency of the synthesized views generated by models like Zero123 directly impact the final reconstruction quality. As observed, inconsistencies between predicted views can introduce artifacts and limit the achievable level of detail. Addressing this bottleneck—enhancing the robustness and 3D coherence of the synthesized multi-view inputs—forms the primary motivation for the work presented in the subsequent chapter.

CHAPTER 3

Multi-View Consistency and Fidelity in 3D Generation

While the One-2-3-45 framework presented in the previous chapter demonstrated the potential of leveraging 2D diffusion priors for rapid single-image 3D reconstruction, its final output quality is inherently coupled with the consistency of the intermediate multi-view predictions synthesized by Zero123. Empirical results revealed that inconsistencies across these generated views lead to compromised 3D reconstruction results.

To address this critical limitation, this chapter introduces One-2-3-45++, a novel method that effectively overcomes the shortcomings of One-2-3-45, delivering significantly enhanced robustness and quality. Taking a single image of any object as input, One-2-3-45++ inherits two primary stages: 2D multi-view generation and 3D reconstruction. During the initial phase, rather than employing Zero123 to predict each view independently, One-2-3-45++ predicts consistent multi-view images jointly. This is realized by tiling a concise set of six-view images into a single image and then finetuning a 2D diffusion model to generate this combined image conditioned on the input reference image. In this way, the 2D diffusion net is able to attend to each view during generation, ensuring more consistent results across views. In the second stage, One-2-3-45++ employs a multi-view conditioned 3D-diffusion-based module to predict the textured mesh in a coarse-to-fine fashion. The consistent multi-view conditional images act as a blueprint for 3D reconstruction, facilitating a zero-shot hallucination capability. Concurrently, the 3D diffusion network excels in lifting the multi-view images, thanks to its ability to harness a broad spectrum of priors extracted from the 3D dataset. Ultimately, One-2-3-45++ employs a lightweight optimization technique to enhance the texture quality efficiently,

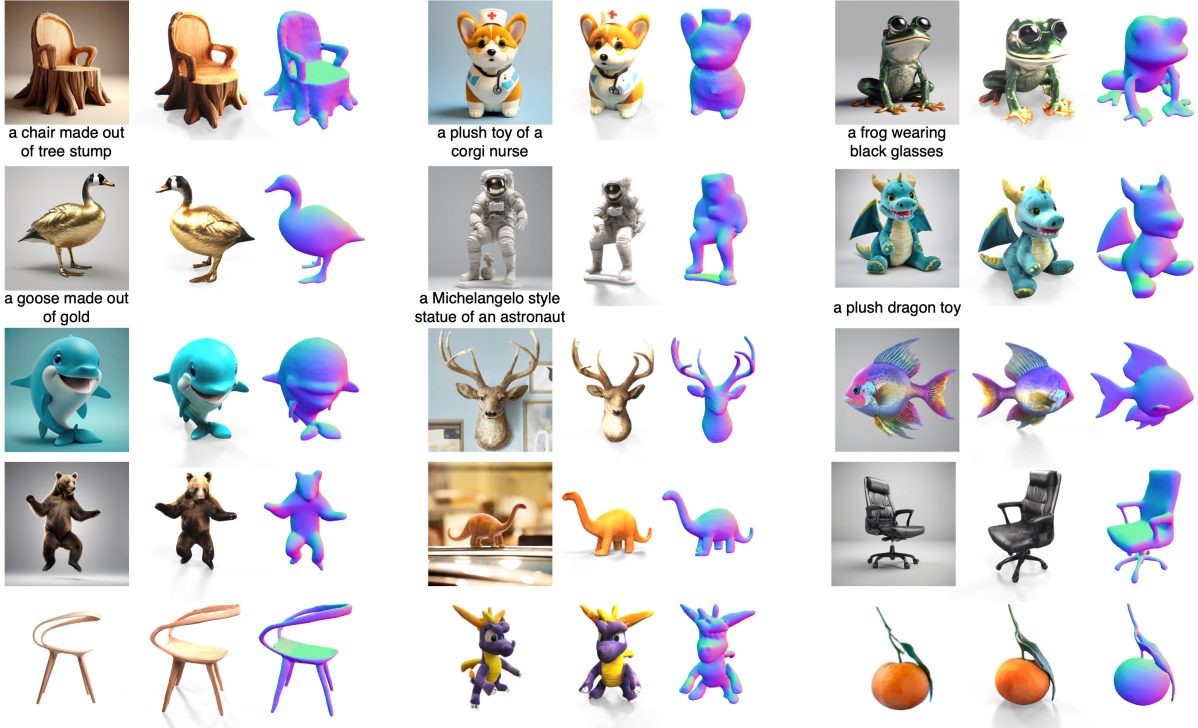


Figure 3.1: One-2-3-45++ is capable of transforming a single RGB image of any object into a high-fidelity textured mesh **in under a minute**. The generated meshes closely mirror the input image. Input image (and text prompt), textured mesh, and normal map are shown.

leveraging the consistent multi-view images for supervision.

As depicted in Fig. 3.1, One-2-3-45++ efficiently generates 3D meshes with realistic textures in under a minute, offering precise fine-grained control. Our comprehensive evaluations, including user studies and objective metrics across an extensive test set, highlight One-2-3-45++’s superiority in terms of robustness, visual quality, and, most importantly, fidelity to the input image.

3.1 Related Work

3.1.1 3D Generation

3D generation has garnered significant attention in recent years. Before the advent of large-scale pre-trained 2D models, researchers often delved into 3D native generative

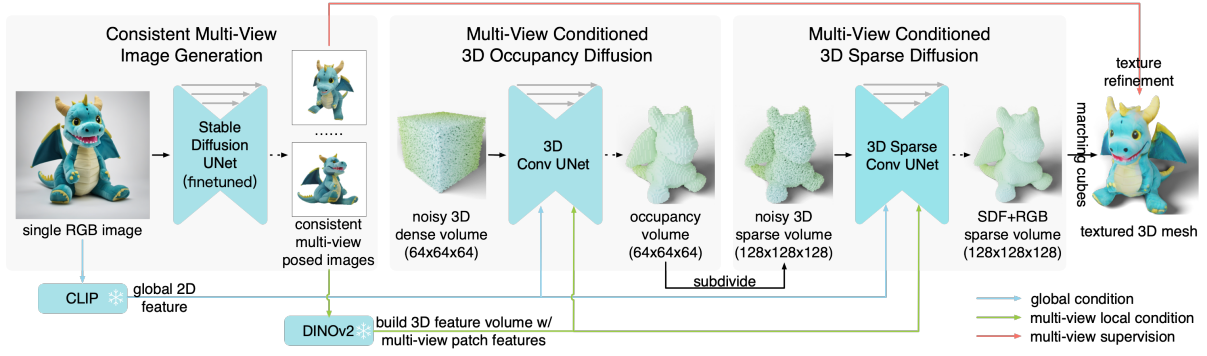


Figure 3.2: Starting with a single RGB image as input, we initially produce consistent multi-view images by fine-tuning a 2D diffusion model. These multi-view images are then elevated into 3D through a pair of 3D native diffusion networks. Throughout the 3D diffusion process, the generated multi-view images act as essential guiding conditions. After extracting the 3D mesh from the denoised volume, we further enhance the texture by employing a lightweight optimization with multi-view images as supervision. One-2-3-45++ is capable of producing an initial textured mesh **within 20 seconds** and delivering a refined one in **roughly one minute** using a single A100 GPU.

models that learn directly from 3D synthetic data or real scans and generate various 3D representations such as point clouds (Achlioptas et al., 2018; Fan et al., 2017; Melas-Kyriazi et al., 2023c; Nichol et al., 2022; Zeng et al., 2022), 3D voxels (Choy et al., 2016; Sanghi et al., 2022; Wu et al., 2017; Xie et al., 2019), polygon meshes (Gao et al., 2022; Groueix et al., 2018; Kanazawa et al., 2018; Nash et al., 2020; Liu et al., 2021; Wang et al., 2018; Liu et al., 2023g), parametric models (Hong et al., 2022), and implicit fields (Jun and Nichol, 2023; Cheng et al., 2023; Gupta et al., 2023; Mescheder et al., 2019; Park et al., 2019; Wu et al., 2023b; Xu et al., 2019; Li et al., 2023b; Zhang et al., 2023; Zheng et al., 2023; Erkoç et al., 2023; Zhao et al., 2023; Yu et al., 2023b). However, given the limited availability of 3D data, these models tended to focus on a select number of categories (e.g., chairs, cars, planes, humans, etc.), struggling to generalize to unseen categories in the open world.

The advent of recent 2D generative models (e.g., DALL-E (Ramesh et al., 2021), Imagen (Saharia et al., 2022), and Stable Diffusion (Rombach et al., 2022)) and vision-language models (e.g., CLIP (Radford et al., 2021)) has equipped us with powerful priors about our 3D world, consequently fueling a surge of research in 3D generation. Notably, models like DreamFusion (Poole et al., 2022), Magic3D (Lin et al., 2023b),

and ProlificDreamer (Wang et al., 2023d) have pioneered a line of approach for per-shape optimization (Jain et al., 2022; Melas-Kyriazi et al., 2023a; Deng et al., 2023; Mohammad Khalid et al., 2022; Lee and Chang, 2022; Metzger et al., 2023; Michel et al., 2022; Raj et al., 2023; Seo et al., 2023; Tang et al., 2023c; Wang et al., 2023a; Xu et al., 2023a,b; Tang et al., 2023b; Chen et al., 2023b; Qian et al., 2023; Chen et al., 2023c; Yu et al., 2023a). These models are designed to optimize a 3D representation for each unique input text or image, drawing on the 2D prior models for gradient guidance. While they have yielded impressive results, these methods tend to suffer from prolonged optimization times, the "multi-face problem," oversaturated colors, and a lack of diversity in results. Some works also concentrate on creating textures or materials for input meshes, utilizing the priors of 2D models (Chen et al., 2023a; Richardson et al., 2023).

A new wave of studies, highlighted by works like Zero123 (Liu et al., 2023d), has showcased the promise of using pre-trained 2D diffusion models for synthesizing novel views from singular images or texts, opening new doors for 3D generation. For instance, One-2-3-45 (Liu et al., 2023b), using multi-view images predicted by Zero123, can produce a textured 3D mesh in a mere 45 seconds. Nevertheless, the multi-view images produced by Zero123 lack 3D consistency. Our research, along with several concurrent studies (Shi et al., 2023c; Liu et al., 2023e; Weng et al., 2023; Ye et al., 2023; Long et al., 2024), is dedicated to enhancing the consistency of these multi-view images – a vital step for subsequent 3D reconstruction applications.

3.1.2 Sparse View Reconstruction

While traditional 3D reconstruction methods, such as multi-view stereo or NeRF-based techniques, often demand a dense collection of input images for accurate geometry inference, many of the latest generalizable NeRF solutions (Chen et al., 2021; Johari et al., 2022; Kulhánek et al., 2022; Liu et al., 2022a; Long et al., 2022; Ren et al., 2023; Trevithick and Yang, 2021; Wang et al., 2022b, 2021b; Yang et al., 2023) strive to learn priors across scenes. This enables them to infer NeRF from a sparse set of images and

generalize to novel scenes. These methods typically ingest a few source views as input, leveraging 2D networks to extract 2D features. These pixel features are then unprojected and aggregated into 3D space, facilitating the inference of density (or SDF) and colors. However, these methods may either rely on consistent multi-view images with accurate correspondences or possess limited priors to generalize beyond training datasets.

Recently, some methods (Zhou and Tulsiani, 2023; Chan et al., 2023; Tewari et al., 2023; Karnewar et al., 2023) have employed diffusion models to aid sparse view reconstruction tasks. However, they generally frame the problem as novel view synthesis, necessitating additional processing, such as distillation using a 3D representation, to generate 3D content. Our work utilizes a multi-view conditioned 3D diffusion model for 3D generation. This model directly learns priors from 3D data and obviates the need for additional post-processing. Moreover, some concurrent works (Long et al., 2024; Liu et al., 2023e; Shi et al., 2023c) employ NeRF-based per-scene optimization for reconstruction, leveraging specialized loss functions.

3.2 Method: One-2-3-45++

In traditional game studios, the creation of 3D content encompasses a series of stages, including concept art, 3D modeling, and texturing, etc. Each stage demands distinct and complementary expertise. For instance, concept artists should possess creativity, a vivid imagination, and the skill to visualize 3D assets. In contrast, 3D modelers must be skilled in 3D modeling tools and capable of interpreting and translating multi-view concept drawings into life-like models, even when drawings contain inconsistencies or errors.

One-2-3-45++ aims to harness the rich 2D priors and the valuable yet limited 3D data following a similar philosophy. As shown in Fig. 3.2, with a single input image of an object, One-2-3-45++ starts by generating coherent multi-view images of the object. This is achieved by finetuning a pre-trained 2D diffusion model and acts akin to the role of a concept artist. These generated images are then input into a multi-view conditioned

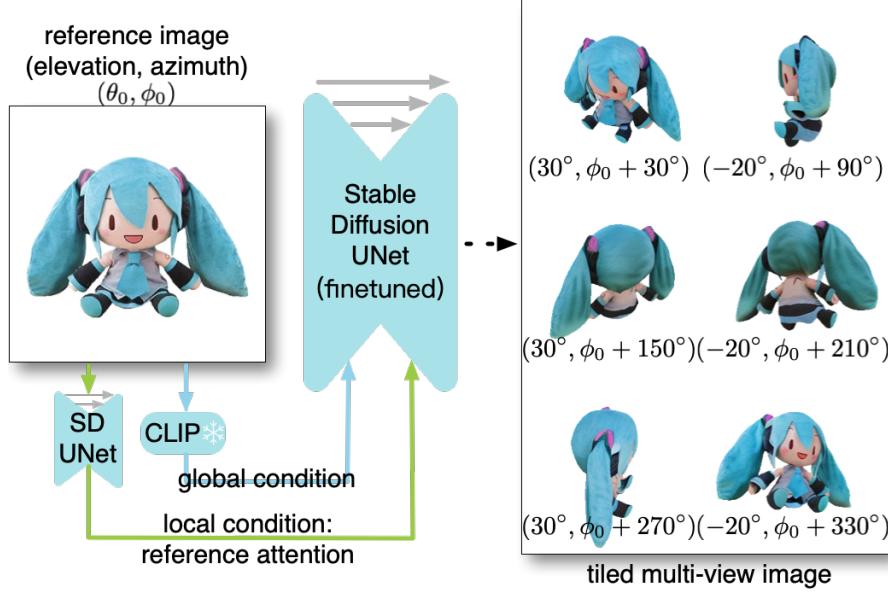


Figure 3.3: **Consistent multi-view generation:** We stitch multi-view images into a single frame and fine-tune the Stable Diffusion model to generate this composite image, using the input reference image as conditions. We utilize predetermined absolute elevation angles and relative azimuth angles. During 3D reconstruction, we do not need to infer the elevation angle of the input image.

3D diffusion model for 3D modeling. The 3D diffusion module, trained on extensive multi-view and 3D pairings, excels at converting multi-view images into 3D meshes. Finally, the produced meshes undergo a lightweight refinement module, guided by the multi-view images, to further enhance the texture quality.

3.2.1 Consistent Multi-View Generation

Recently, Zero123 has demonstrated the potential of fine-tuning a pretrained 2D diffusion network to incorporate camera view control, thereby synthesizing novel views of an object from a single reference image. While previous studies have employed Zero123 to generate multi-view images, they often suffer from inconsistencies across different views. This inconsistency arises because Zero123 models the conditional marginal distribution for each view in isolation, without considering inter-view communication during multi-view generation. In this work, we present an innovative method to produce consistent multi-view images, significantly benefiting downstream 3D reconstruction.

Multi-View Tiling To generate multiple views in a single diffusion process, we adopt a simple strategy by tiling a sparse set of 6 views into a single image with a 3×2 layout as shown in Fig. 3.3. Subsequently, we finetune a pre-trained 2D diffusion net to generate the composite image, conditioned on a single input image. This strategy enables multiple views to interact with each other during the diffusion.

It’s nontrivial to define the camera poses of the multi-view images. Given that the 3D shapes within the training dataset lack aligned canonical poses, employing absolute camera poses for the multi-view images could lead to ambiguities for the generative model. Alternatively, if we were to set the camera poses relative to the input view, as done in Zero123, downstream applications would then be required to infer the elevation angle of the input image to deduce the camera poses of the multi-view images. This additional step could introduce errors into the pipeline. To address these, we opt for fixed absolute elevation angles paired with relative azimuth angles to define the poses of multi-view images, effectively resolving the orientation ambiguity without necessitating further elevation estimation. To be more precise, the six poses are determined by alternating elevations of 30° and -20° , coupled with azimuths commencing at 30° and incrementing by 60° for each subsequent pose, as shown in Fig. 3.3.

Network and Training Details To fine-tune Stable Diffusion for adding image conditioning and generating coherent multi-view composite images, we employ three crucial network or training designs: (a) **Local Condition:** We adopt the reference attention technique (Zhang, 2023) to incorporate the local condition of the image patch features. Specifically, we process the reference input image with the denoising UNet model and append the self-attention key and value matrices of the image tokens from the conditional reference image to the corresponding attention layers of the denoising multi-view image. (b) **Global Condition:** We leverage CLIP image embedding as a global condition, by replacing the text token features originally used in Stable Diffusion with the duplicated CLIP image features. These global image embeddings are multiplied by a set of learnable weights, providing the network with an overall semantic understanding of the object. (c) **Noise Schedule:** The original Stable Diffusion model was trained using a scaled-linear

noise schedule. We found it necessary to switch to a linear noise scheme in our fine-tuning process.

We fine-tune the Stable Diffusion2 *v*-mode using 3D shapes from the Objaverse dataset (Deitke et al., 2023b). For each shape, we generate three data points by randomly sampling the camera pose of the input image from a specified range, and selecting a random HDRI environment lighting from a curated set that offers uniform lighting. Initially, we fine-tuned only the self-attention layers along with the key and value matrices of the cross-attention layers using LoRA (Hu et al., 2021). Subsequently, we fine-tuned the entire UNet using a conservative learning rate. The finetuning process was conducted using 16 A100 GPUs and took approximately 10 days.

3.2.2 3D Diffusion with Multi-View Condition

While prior work utilizes generalizable NeRF methods for 3D reconstruction, it primarily depends on accurate local correspondence of multi-view images and possesses limited priors for 3D generation. This constrains their effectiveness in lifting intricate and inconsistent multi-view images generated by the 2D diffusion network. Instead, we propose an innovative way to lift the generated multi-view images to 3D by utilizing a multi-view conditioned 3D generative model. It seeks to learn a manifold of plausible 3D shapes conditioned on multi-view images by training expressive 3D native diffusion networks on extensive 3D data.

3D Volume Representations As shown in Fig. 3.2, we represent a textured 3D shape as two discrete 3D volumes, a signed distance function (SDF) volume, and a color volume. The SDF volume measures the signed distance from the center of each grid cell to the nearest shape surface, while the color volume captures the color of the closest surface points relative to the center of the grid cells. Additionally, we generate a discrete occupancy volume for the 3D shape, where each grid cell stores a binary occupancy based on whether the absolute value of its SDF is below a predefined threshold. The occupancy volume depicts the shell of the 3D shape.

Two-Stage Diffusion Capturing fine-grained details of 3D shapes necessitates the use of high-resolution 3D grids, which unfortunately entail substantial memory and computational costs. We thus follow LAS-Diffusion (Zheng et al., 2023) to generate high-resolution volumes in a coarse-to-fine two-stage manner. Specifically, the initial stage generates a low-resolution (e.g., $n = 64$) full 3D occupancy volume $F \in \mathbb{R}^{n \times n \times n \times 1}$ to approximate the shell of the 3D shape. The second stage then focuses on the occupied shell region only and aims to generate a high-resolution (e.g., 128^3) four-channel sparse volume S , which predicts fine-grained SDF values and color for the sparsely occupied shell region.

We employ a separate diffusion network for each stage. For the first stage, normal 3D convolution is used within the UNet to produce the full 3D occupancy volume F , while for the second stage, we incorporate 3D sparse convolution (Tang et al., 2023a) in the UNet to yield the 3D sparse volume S . Both diffusion networks are trained using the denoising loss (Ho et al., 2020):

$$\mathcal{L}_{x_0} = \mathbb{E}_{\epsilon \sim \mathcal{N}(0, I), t \sim \mathcal{U}(0, 1)} \|f(x_t, t, c) - x_0\|_2^2$$

where ϵ and t are sampled noise and time step, x_0 is a data point (F or S) and x_t is its noised version, c is the multi-view condition, and f is the UNet. $\mathcal{N}$ and $\mathcal{U}$ denote Gaussian and uniform distribution, respectively.

Multi-View Condition Training a conventional 3D native diffusion network can be challenging to generalize due to the limited availability of 3D data. However, the use of generated multi-view images can provide a comprehensive guide, greatly simplifying the imagination difficulty of 3D generation. We integrate the multi-view images to guide the diffusion process by initially extracting local image features and subsequently constructing a conditional 3D feature volume, denoted as C . This strategy follows the rationale that local priors facilitate easier generalization (Zheng et al., 2023).

As shown in Fig. 3.4, given m multi-view images, we first employ a pre-trained 2D backbone, DINOv2, to extract a set of local patch features for each image. We then build

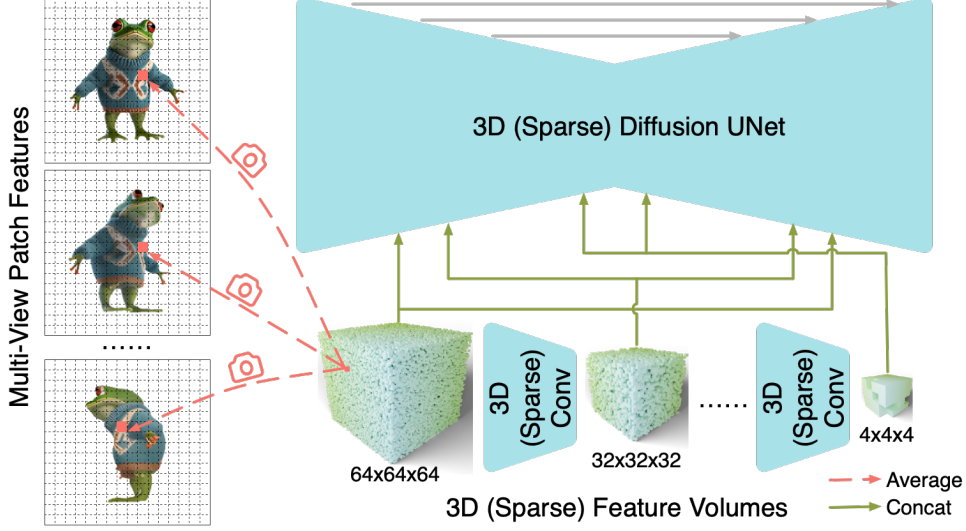


Figure 3.4: **Multi-view local condition:** We employ a pre-trained 2D backbone to extract 2D patch features for each view. These features are then aggregated using known projection matrices to construct a 3D feature volume. The volume is further processed by 3D convolutional neural networks, resulting in feature volumes of varying resolutions. Subsequently, these volumes are concatenated with the corresponding feature volumes within the diffusion U-Net to guide the 3D diffusion.

a 3D feature volume C by projecting each 3D voxel within the volume onto m multi-view images using the known camera poses. For each 3D voxel, we aggregate m associated 2D patch features through a shared-weight MLP, followed by max pooling. These aggregated features collectively form the feature volume C .

In the diffusion network, the UNet consists of several levels. For example, the occupancy UNet in the initial stage has five levels: 64^3 , 32^3 , 16^3 , 8^3 , and 4^3 . Initially, we construct a conditional feature volume C that matches the starting resolution, as outlined earlier. A 3D convolution network is then applied to C , producing volumes for the subsequent resolutions. The resultant conditional volumes are then concatenated with the volumes inside the UNet to guide the diffusion process. For the second stage, we construct sparse conditional volumes and utilize 3D sparse convolution. To benefit the diffusion of color volume, we also concatenate 2D pixel-wise projected colors to the final layer of the diffusion UNet. Moreover, we integrate the CLIP feature of the input image as a global condition.

Training and Inference Details We train the two diffusion networks using 3D shapes

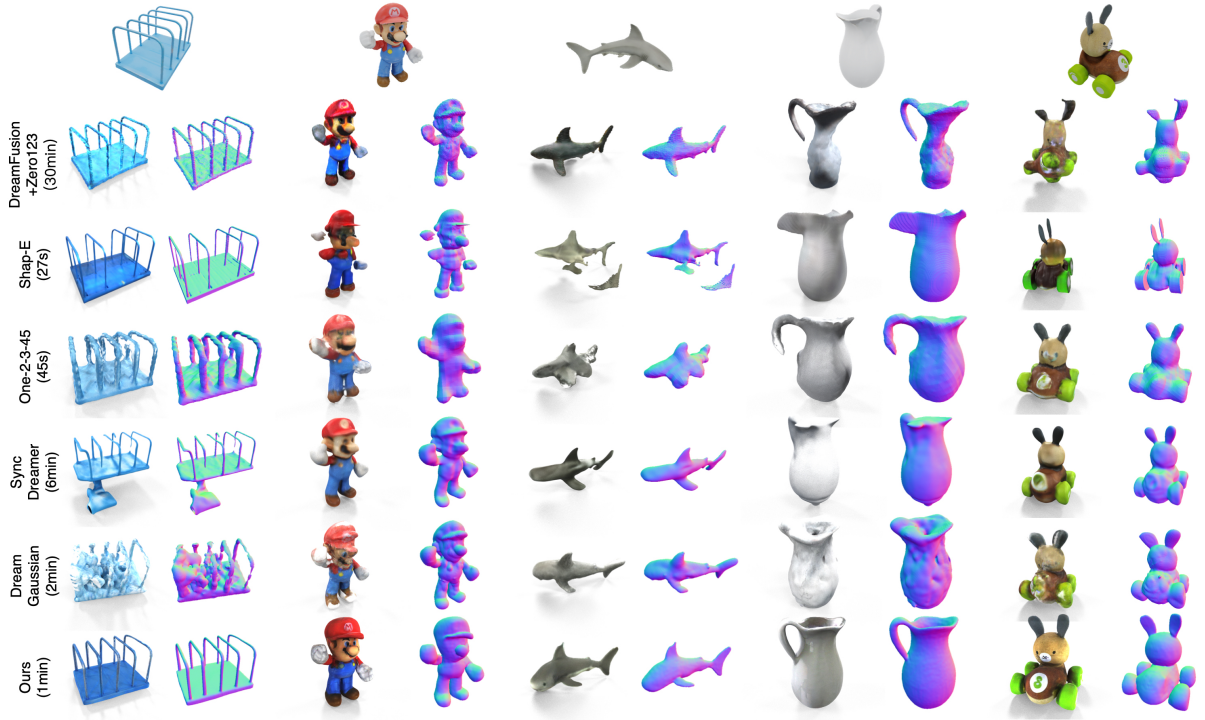


Figure 3.5: Qualitative results of various single image to 3D approaches. Input images, textured meshes, and normal maps are shown.

from the Objaverse dataset (Deitke et al., 2023b). For each 3D shape, we first convert it to a watertight manifold before extracting its SDF volume. We unproject the multi-view renderings of the shape to get a 3D colored point cloud, which is used to build the color volume. During training, we utilize the ground truth renderings to serve as the multi-view conditions. Since two diffusion networks are trained separately, we introduced random perturbations to camera poses and infused random noises to the initial occupancy of the second stage to enhance robustness. We train the two diffusion nets using 8 A100 GPUs for about 10 days for each stage.

During inference, a 64^3 grid is first initialized with Gaussian noise and then denoised by the first diffusion net. Each predicted occupied voxel is further subdivided into 8 smaller voxels, used to construct a high-resolution sparse volume. The sparse volume is initialized with Gaussian noise and then denoised with the second diffusion net, resulting in predictions for the SDF and color of each voxel. The Marching Cubes algorithm is finally applied to extract a textured mesh.

Table 3.1: Comparison on single image to 3D. Evaluated on the GSO (Downs et al., 2022) dataset, which contains 1,030 3D objects.

Method	F-Score (%)↑	CLIP-Sim↑	User-Pref. (%)↑	Time↓
Zero123 XL (Deitke et al., 2023a)	91.6	73.1	58.6	30min
One-2-3-45 (Liu et al., 2023b)	90.4	70.8	52.7	45s
SyncDreamer (Liu et al., 2023e)	84.8	68.9	28.4	6min
DreamGaussian (Tang et al., 2023b)	81.0	68.4	31.5	2min
Shap-E (Jun and Nichol, 2023)	91.8	73.1	40.8	27s
Ours	93.6	81.0	87.6	60s

3.2.3 Texture Refinement

Given that multi-view images possess higher resolution than the 3D color volume, we can refine the texture of the generated mesh through a lightweight optimization process. To achieve this, we fix the geometry of the generated mesh while optimizing a color field represented by a TensorRF (Chen et al., 2022). In each iteration, the generated 3D mesh is rasterized, and the color network is queried to produce 2D renderings. We leverage the predicted consistent multi-view images to guide the texture optimization using a l_2 loss. Lastly, we bake the optimized color field onto the mesh, with the surface normal serving as the viewing direction.

3.3 Experiments

3.3.1 Comparison on Image to 3D

Baselines: We evaluate One-2-3-45++ against both optimization-based and feed-forward methods. Within the optimization-based approaches, our baselines include DreamFusion (Poole et al., 2022) with Zero123 XL (Liu et al., 2023d) as its backbone, as well as SyncDreamer (Liu et al., 2023e), and DreamGaussian (Tang et al., 2023b). For feed-forward approaches, we compare with One-2-3-45 (Liu et al., 2023b) and Shap-E (Jun and Nichol, 2023). We employ the ThreeStudio (Guo et al., 2023) implementation for Zero123 XL (Guo et al., 2023) and the original official implementations for the other methods.

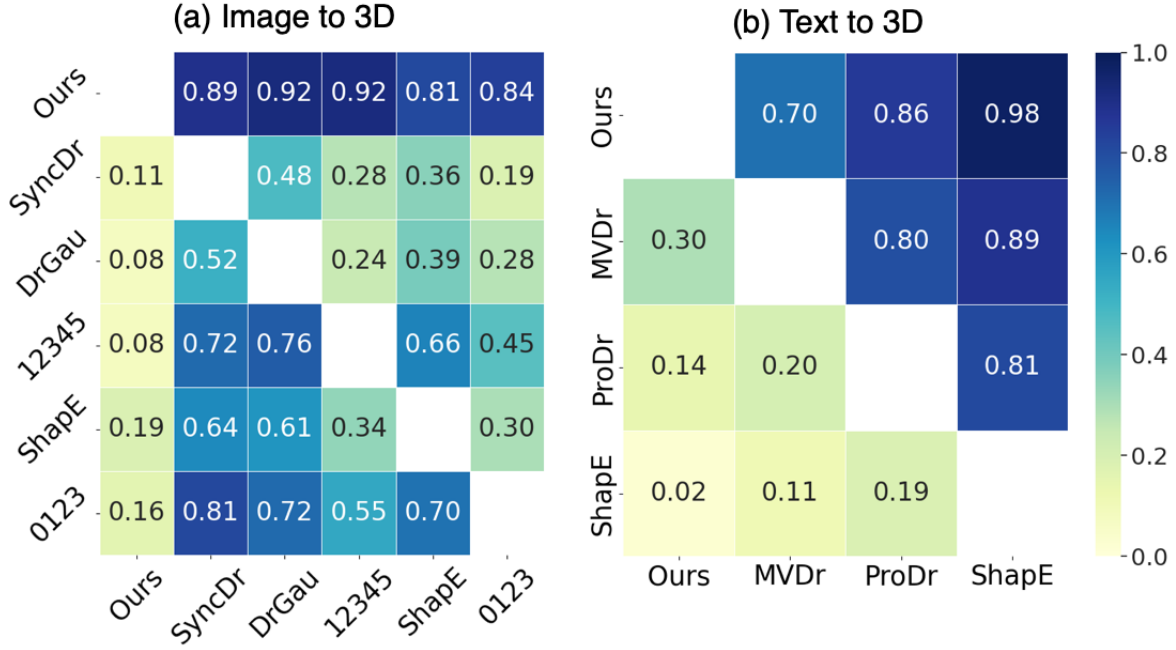


Figure 3.6: Results of a user study involving 53 participants. Each cell displays the probability or preference rate at which one method (row) outperforms another (column).

Dataset and Metrics: We assess the performance of the methods using the entire set of 1,030 shapes from the GSO dataset (Downs et al., 2022), which were not exposed to any of the methods during training to the best of our knowledge. For each shape, we generate a frontal view image to serve as the input. In line with One-2-3-45 (Liu et al., 2023b), we employ the F-Score and CLIP similarity as our evaluation metrics. The F-Score evaluates the geometric similarity between the predicted mesh and the ground truth mesh. For the CLIP similarity metric, we render 24 different views for each predicted and ground truth mesh, compute the CLIP similarity for each corresponding pair of images, and then average these values across all views. Prior to metric computation, we align the predicted mesh with the ground truth mesh using a combination of linear search and the ICP algorithm.

User Study: A user study was also carried out. For each participant, 45 shapes were randomly selected from the entire GSO dataset, and two methods were randomly sampled for each shape. Participants were asked to choose the result from each pair of comparative outcomes that exhibits superior quality and better aligns with the input image. The

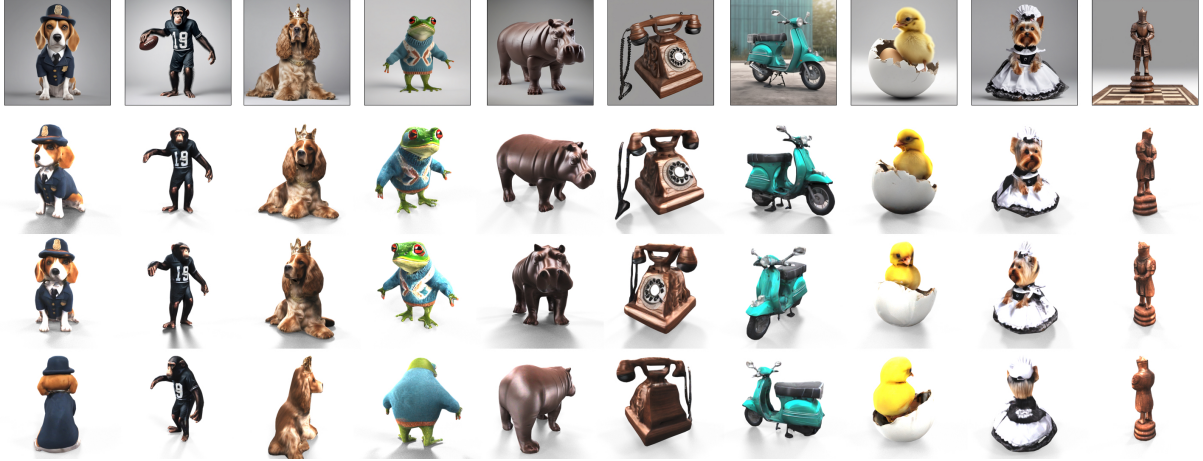


Figure 3.7: Our qualitative results: top row displays input images; subsequent rows show multi-view renderings of the generated meshes.

preference rate for all methods was then tallied based on these selections. In total, 2,385 evaluated pairs were collected from 53 participants.

Results: As presented in Table 3.1, One-2-3-45++ surpasses all baseline methods regarding F-Score and CLIP similarity. The user preference scores further highlight a significant performance disparity, with our method outperforming competing approaches by a substantial margin. Refer to Fig. 3.6 for an in-depth confusion matrix, which illustrates that One-2-3-45++ outperforms One-2-3-45 92% of the time. Moreover, when compared to optimization-based methods, our approach demonstrates notable runtime advantages. Fig. 3.5 and 3.7 show qualitative results.

3.3.2 Comparison on Text to 3D

Baselines: We compared One-2-3-45++ with optimization-based methods, specifically ProlificDreamer (Wang et al., 2023d) and MVDream (Shi et al., 2023c), as well as a feed-forward approach, Shap-E (Jun and Nichol, 2023). For ProlificDreamer, we utilized the ThreeStudio implementation (Guo et al., 2023), while for the remaining methods, we employed their respective official implementations.

Dataset and Metrics: Given that many baseline approaches necessitate hours to produce a single 3D shape, our evaluation was conducted on 50 text prompts, sampled



Figure 3.8: Qualitative results of various text-to-3D approaches. Input images, textured meshes, and normal maps are shown.

Table 3.2: Quantitative comparison with various text to 3D methods. Evaluated on 50 text prompts from DreamFusion (Poole et al., 2022).

Method	CLIP-Sim $\uparrow$	User-Pref. $\uparrow$	Runtime $\downarrow$
ProlificDreamer (Wang et al., 2023d)	25.7	39.5	10h+
MVDream (Shi et al., 2023c)	24.8	66.2	2h
Shap-E (Jun and Nichol, 2023)	22.3	11.1	27s
Ours	26.8	84.1	60s

from DreamFusion (Poole et al., 2022). We utilize CLIP similarity, calculated by comparing 24 rendered views of the predicted mesh against the input text prompt and then averaging the similarity scores across all views.

User Study: The user study, akin to the image-to-3D evaluation, involved 30 pairs of outcomes randomly selected for each participant. In total, 1,590 evaluation pairs were collected from 53 participants.

Results: As illustrated in Table 3.2, One-2-3-45++ outperforms all baseline methods in terms of CLIP similarity. This is further corroborated by user preference scores, with our method significantly outshining rival techniques. See Fig. 3.6 for an in-depth analysis. When directly comparing One-2-3-45++ with the second-best method, MVDream (Shi et al., 2023c), our approach commands a 70% user preference rate. Moreover, while our method delivers prompt results, MVDream (Shi et al., 2023c) requires about 2 hours to

Table 3.3: Ablation studies of different modules. Evaluated on the complete GSO (Downs et al., 2022) dataset. “MultiView”, “Reconstruction”, and “Texture” indicate multi-view generation, sparse view reconstruction, and texture refinement modules, respectively.

MultiView	Reconstruction	Texture	F-Score↑	CLIP-Sim↑	Time↓
Zero123 XL (Deitke et al., 2023a)	Ours	w/o	92.9	71.9	14s
Ours	SparseNeuS (Long et al., 2022)	w/o	81.2	67.2	15s
Ours	Ours	w/o	93.6	73.4	20s
Ours	Ours	w/	93.6	81.0	60s

generate a single shape. Fig. 3.8 shows qualitative results.

3.3.3 Analyses

Ablation Studies of Overall Pipeline One-2-3-45++ is comprised of three key modules: consistent multi-view generation, multi-view conditioned 3D diffusion, and texture refinement. We conducted ablation studies on these modules using the complete GSO dataset (Downs et al., 2022), with results detailed in Table 3.3. Replacing our consistent multi-view generation module with Zero123XL (Deitke et al., 2023a) led to a noticeable performance decline. Furthermore, substituting our 3D diffusion module with the generalizable NeRF used in One-2-3-45 (Liu et al., 2023b) resulted in an even more significant performance drop. However, the inclusion of our texture refinement module markedly improved texture quality, yielding higher CLIP similarity scores.

Ablation Studies of 3D Diffusion Table 3.4 presents the results of an ablation study of the 3D diffusion module. The study highlights the importance of multi-view images for the module’s efficacy. When the module operates without multi-view conditions, relying solely on the global CLIP feature from a single input view (rows a and f), there is a significant decline in performance. Conversely, the One-2-3-45++ approach leverages multi-view local features by constructing a 3D feature volume with known projection matrices. A mere concatenation of global CLIP features from multiple views also impairs performance (rows b and f), underlining the value of multi-view local conditions. Global CLIP features of the input view, however, provide global shape semantics; their removal results in decreased performance (rows c and e). Although One-2-3-45++ uses predicted multi-view

Table 3.4: Ablation study of the 3D diffusion module. 3D IoU of the initial-stage occupancy prediction is reported. Note that the 3D IoU is computed for the 3D shell, excluding the solid interior.

id	multi-view cond.	global cond.	image source	proj. perturb.	3D IoU $\uparrow$
a	w/o	w/	rendering	N/A	18.3
b	global	w/	rendering	N/A	24.4
c	local	w/o	rendering	w/o	41.4
d	local	w/	prediction	w/o	41.9
e	local	w/	rendering	w/o	44.1
f	local	w/	rendering	w/	45.1

images for 3D reconstruction, incorporating these predicted images during training of the 3D diffusion module can lead to a performance downturn (rows d and e) due to the potential mismatch between the predicted multi-view images and actual 3D ground truth meshes. To train the module effectively, we utilize ground truth renderings. Recognizing that predicted multi-view images may be flawed, we introduce random perturbations to projection matrices during training to enhance robustness when processing predicted multi-view images (rows e and f).

Comparison on Multi-View Generation We also evaluate our consistent multi-view generation module against existing approaches, namely Zero123 (Liu et al., 2023d) and its scaled variant (Deitke et al., 2023a), alongside two concurrent works: SyncDreamer (Liu et al., 2023e) and Wonder3D (Long et al., 2024). Our comparison utilizes the GSO (Downs et al., 2022) dataset, where for each object, we render a single input image and task the methods with producing multi-view images. For Zero123 and Zero123 XL, we utilize the same target poses as our approach. However, for Wonder3D and SyncDreamer, we employ the target poses preset by these methods, as they do not support altering camera positions during inference. As presented in Table 3.5, our approach surpasses current methodologies in PSNR, LPIPS, and foreground mask IoU. Notably, Wonder3D (Long et al., 2024) employs orthographic projection in its training phase, which compromises its robustness when dealing with perspective images during inference. SyncDreamer (Liu et al., 2023e) only generates views at an elevation of 30° , a simpler setting than ours.

Table 3.5: Comparison of different multi-view generation methods. Evaluated on the complete GSO (Downs et al., 2022) dataset.

	Target Elevations	PSNR $\uparrow$	LPIPS $\downarrow$	Mask IoU $\uparrow$
Zero123 (Liu et al., 2023d)	30° and −20°	20.32	0.110	0.856
Zero123 XL (Deitke et al., 2023a)		20.11	0.113	0.869
Ours		22.12	0.110	0.878
SyncDreamer (Liu et al., 2023e)	30°	21.67	0.095	0.894
Wonder3D (Long et al., 2024)	0°	18.67	0.130	0.635

3.4 Discussion

In this chapter, we introduced One-2-3-45++, an innovative approach for transforming a single image of any object into a 3D textured mesh. This method stands out by offering more precise control compared to existing text-to-3D models, and it is capable of delivering high-quality meshes swiftly—typically in under a minute. Additionally, the generated meshes exhibit a high fidelity to the original input image.

Looking ahead, there is potential to enhance the robustness and detail of the geometry by incorporating additional guiding conditions from 2D diffusion models, alongside RGB images. Alternative future strategies might also explore fundamentally different structured or decomposed latent 3D representations, potentially offering distinct advantages in controllability and editability compared to the current volumetric diffusion approach.

Moreover, the methods discussed thus far primarily address the challenge of generating 3D objects from a single input view. A crucial next step towards truly generalizable 3D perception is handling even more challenging scenarios involving sparse, potentially unposed input views. Tackling the inherent ambiguity and lack of correspondence in such sparse data necessitates better strategies for understanding the spatial relationships between views, a challenge we turn to in the following chapter.

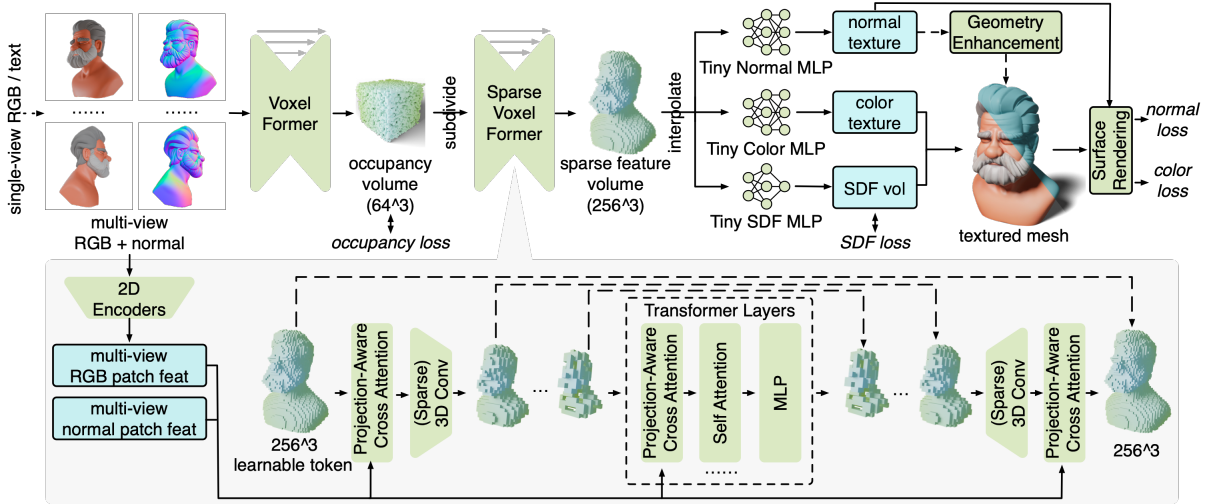


Figure 3.9: **Pipeline Overview.** MeshFormer takes a sparse set of multi-view RGB and normal images as input, which can be estimated using existing 2D diffusion models. We utilize a 3D feature volume representation, and submodules Voxel Former and Sparse Voxel Former share a similar novel architecture, detailed in the gray region. We train MeshFormer in a unified single stage by combining mesh surface rendering and 512^3 SDF supervision. MeshFormer learns an additional normal texture, which can be used to further enhance the geometry and generate fine-grained sharp geometric details.

3.5 Related Project

Above, we have discussed One-2-3-45++, which demonstrates the effectiveness of using two-stage 3D diffusion models conditioned on generated consistent multi-view images. However, it still struggles to generate high-quality textures or fine-grained geometry. In the related project below, we propose a redesigned reconstruction model with significantly increased capacity, achieving superior mesh quality and detail.

3.5.1 MeshFormer

While open-world 3D reconstruction models have recently garnered significant attention, achieving high-quality mesh outputs efficiently often remains a challenge. Many existing methods, without sufficient 3D inductive bias, entail expensive training costs and can struggle to extract clean, detailed 3D meshes. In this work, we introduce MeshFormer, a sparse-view reconstruction model that explicitly leverages 3D native structure, input

guidance, and training supervision. Specifically, instead of using a triplane representation, we store features in 3D sparse voxels and combine transformers with 3D convolutions to leverage an explicit 3D structure and projective bias. In addition to sparse-view RGB input, we require the network to take input and generate corresponding normal maps. The input normal maps can be predicted by 2D diffusion models, significantly aiding in the guidance and refinement of the geometry’s learning. Moreover, by combining Signed Distance Function (SDF) supervision with surface rendering, we directly learn to generate high-quality meshes without the need for complex multi-stage training processes. By incorporating these explicit 3D biases, MeshFormer can be trained efficiently and deliver high-quality textured meshes with fine-grained geometric details. It can also be integrated with 2D diffusion models to enable fast single-image-to-3D and text-to-3D tasks.

The pipeline of the system can be found in Fig. 3.9. For more details, please refer to (Liu et al., 2024b).

CHAPTER 4

Sparse-View 3D Reconstruction and Pose Estimation

3D object reconstruction is a long-standing problem with applications that span 3D content creation, augmented reality, virtual reality, and robotics, among others. Although traditional photogrammetry (Barnes et al., 2009; Schönberger et al., 2016; Stereopsis, 2010) and recent neural field methods (Mildenhall et al., 2021; Wang et al., 2021a; Yariv et al., 2021) have made significant strides in reconstructing high-fidelity geometry and appearance, they typically require dense view inputs. However, in many practical scenarios, such as in e-commerce and consumer capture situations, acquiring a comprehensive set of high-resolution images along with precise camera data is not always feasible.

On the other end of the spectrum, as discussed in the previous chapters, the tasks of converting a single image to 3D and text to 3D have recently seen substantial progress (Poole et al., 2022; Lin et al., 2023b; Li et al., 2023a; Liu et al., 2024a; Shi et al., 2023c; Wang et al., 2023d), thanks to the rich priors embedded in 2D diffusion models (Ramesh et al., 2021; Saharia et al., 2022; Rombach et al., 2022) and pre-training on extensive 3D datasets (Deitke et al., 2023b). These methods may achieve high-quality geometry and texture that matches the input view, but they also introduce ambiguities in the regions not visible in the input image (such as the back view). Although these methods attempt to hallucinate reasonable interpretations of these invisible areas, the generated regions may not always align with users’ expectations, and users often lack sufficient control over these ambiguous regions.

In this chapter, we explore a critical scenario in which the input consists of one or a few unposed 2D images of a single object. The images are captured from arbitrarily distributed camera poses, often with little to no overlap. We tackle both the 3D reconstruction and

pose estimation of input images under this sparse view setting. Note that, in dense view setting, traditional Structure-from-Motion (SfM) solvers (e.g., COLMAP (Schonberger and Frahm, 2016)) are typically employed for pose estimation. However, with sparse view inputs, these solvers often become unreliable and tend to fail due to insufficient overlapping visual cues. This issue is the main reason why existing sparse view reconstruction methods (Kim et al., 2022; Long et al., 2022; Zhou and Tulsiani, 2023) generally require known camera poses as input. While some recent methods have attempted pose-free reconstruction and pose estimation for sparse views (Lin et al., 2023a; Sinha et al., 2023; Zhang et al., 2022a; Jiang et al., 2022, 2023), they are usually trained on a predefined small set of object categories and exhibit poor generalization to unseen object categories.

In response, we propose an innovative class-agnostic approach called SpaRP, capable of processing arbitrary object categories with unposed sparse views. Our inspiration comes from recent breakthroughs in open-domain single-image-to-3D methods. They leverage 2D diffusion models (e.g., Stable Diffusion (Rombach et al., 2022)) to generate novel viewpoints of an object (Liu et al., 2023d), and even consistent multi-view images from a single input image (Shi et al., 2023c,a; Li et al., 2023a; Liu et al., 2023e; Long et al., 2024), by finetuning the diffusion models with corresponding multi-view image pairs. These discoveries imply that 2D diffusion models harbor rich priors concerning 3D objects. Instead of merely producing multi-view images, we contemplate leveraging 2D diffusion models to examine a set of unposed input images from sparse viewpoints, infer their spatial interrelationships, and recover relative camera poses and underlying 3D shapes.

Specifically, we finetune a 2D diffusion model (Rombach et al., 2022) to process sparse input views by compositing them into a single image for conditioning. The diffusion model is concurrently tuned to deduce the relative poses of the input images and the underlying 3D objects. For the relative pose estimation branch, instead of outputting camera poses as scalars, we task 2D diffusion models to produce a surrogate representation: the NOCS maps (Wang et al., 2019) that embed pixel-wise correspondences across different views and are more suitable for 2D diffusion models. From these maps, we extract the relative



Figure 4.1: SpaRP handles open-world 3D reconstruction and pose estimation from unposed sparse-view images, delivering results in approximately 20 seconds.

camera poses for the sparse views using the traditional PnP algorithm (Bradski, 2000), assuming known camera intrinsics. For the reconstruction branch, the diffusion model is tasked to produce multi-view images of the object from fixed known camera poses, covering the entire 3D object. This task requires the models to incorporate all information from input sparse views and hallucinate invisible regions. We then feed the generated images with fixed known poses into a pre-trained 3D reconstruction module (Liu et al., 2024a) to create a textured 3D mesh. We can further refine the estimated camera poses by aligning the input views with the generated mesh through differentiable rendering (Laine et al., 2020).

We train SpaRP on the Objaverse (Deitke et al., 2023b) dataset with 1–6 unposed input views. Unlike some previous methods that rely on costly per-shape optimization (Wu et al., 2023a), our method delivers 3D textured meshes along with camera poses in a much more efficient manner, requiring only ~ 16 seconds. As shown in Fig. 4.1, our approach can faithfully generate 3D assets that closely follow the reference unposed images, effectively overcoming the ambiguity issue of single-image-to-3D. Extensive evaluation on three datasets demonstrates the superior performance of our method over baselines in reconstructing 3D meshes with vivid appearance and high-fidelity geometry, alongside precise pose estimation of the input images.

4.1 Related Work

4.1.1 Sparse-View 3D Reconstruction

Reconstructing 3D objects from sparse-view images is challenging due to the lack of visual correspondence and clues. When a small baseline between images is assumed, several methods (Chen et al., 2021; Long et al., 2022; Rematas et al., 2021; Trevithick and Yang, 2021; Yu et al., 2021a; Johari et al., 2022; Kulhánek et al., 2022; Liu et al., 2022a; Ren et al., 2023; Wang et al., 2022b, 2021b; Yang et al., 2023) have pretrained generalizable models to infer surface positions by establishing pixel correspondences and learning generalizable priors across scenes. However, these methods often fail to produce satisfactory results when the sparse-view images have a large baseline. Some studies have attempted to alleviate the dependence on dense views by incorporating priors or adding regularization (Shi et al., 2023b; Jain et al., 2021; Kim et al., 2022; Niemeyer et al., 2022) into the NeRF optimization process. Others have employed 2D diffusion priors to generate novel-view images as additional input for the NeRF model (Zhou and Tulsiani, 2023; Chan et al., 2023; Tewari et al., 2023; Karnewar et al., 2023). For example, ReconFusion (Wu et al., 2024) trains a NeRF from sparse-view images and uses a denoising UNet to infer some novel view images as support for the NeRF model. EscherNet (Kong et al., 2024) utilizes Stable Diffusion for novel view synthesis and designs a camera positional encoding

module to yield more consistent images. Furthermore, some recent works (Long et al., 2024; Liu et al., 2023e; Shi et al., 2023c) have integrated specialized loss functions and additional modalities as inputs into NeRF-based per-scene optimization.

In contrast to these methods, our approach does not require camera poses for the input sparse views. It is not limited to small baselines and is capable of generating 360-degree meshes. Furthermore, without the need for per-shape optimization, our method can quickly produce both textured meshes and camera poses in about 20 seconds.

4.1.2 Pose-Free Reconstruction

Unlike the methods mentioned above, which assume known camera poses, many studies have aimed to solve the pose-free reconstruction challenge. When provided with dense images, some approaches (Lin et al., 2021; Wang et al., 2021c; Xia et al., 2022) jointly optimize the NeRF representation along with camera parameters. However, due to the highly non-convex nature of this optimization problem, such methods are susceptible to initial pose guesses and can become trapped in local minima. This issue worsens when input images are sparse, with increasing ambiguity and reduced constraint availability. In response, numerous proposals have attempted to enhance optimization robustness. For example, SpaRF (Truong et al., 2023) uses dense image matches as explicit optimization constraints, while FvOR (Yang et al., 2022) starts with coarse predictions of camera poses and alternated updates between shape and pose.

In contrast to the optimization-based methods, there is a body of research proposing generalizable solutions for this problem. VideoAE (Lai et al., 2021) infers scene geometry from the first frame in a video series and estimates camera poses relative to that frame, which allows for warping scene geometry to decode new viewpoints. SparsePose (Sinha et al., 2023) first regresses and then iteratively refines camera poses. FORGE (Jiang et al., 2022) designs neural networks to infer initial camera poses, fuse multi-view features, and decode spatial densities and colors. GRNN (Tung et al., 2019) offers a GRU-based reconstruction method estimating the relative pose for each input view against a global

feature volume. The RelPose series (Zhang et al., 2022a; Lin et al., 2023a) use probabilistic modeling for relative rotation estimation between images. Other works (Sajjadi et al., 2022; Jiang et al., 2023) eschew explicit camera pose estimations, instead employing transformers to encode input views into latent scene representations for novel view synthesis.

More recently, leveraging large vision models and diffusion models, which have shown significant promise, new efforts have emerged for camera pose estimation. PoseDiffusion (Wang et al., 2023b) implements a diffusion model guided by 2D keypoint matches to estimate poses. PF-LRM (Wang et al., 2023c) adapts the LRM model (Hong et al., 2023) to predict a point cloud for each input image, then utilizes differentiable PnP for pose estimation. iFusion (Wu et al., 2023a) employs an optimization pipeline to assess relative elevations and azimuths. It utilizes Zero123 (Liu et al., 2023d) predictions as a basis and optimizes the relative pose between two images by minimizing the reconstruction loss between the predicted and target images.

In contrast to these existing approaches, our proposal capitalizes on the extensive priors inherent in pre-trained 2D diffusion models, thereby providing exceptional generalizability to handle a diverse range of open-world categories. Our method predicts camera poses and 3D mesh geometry in a single feedforward pass, negating the need for per-shape optimization.

4.1.3 Open-World 3D Generation

Open-world single-image-to-3D and text-to-3D tasks have recently undergone significant advancements. Recent 2D generative models (Ramesh et al., 2021; Saharia et al., 2022; Rombach et al., 2022) and vision-language models (Radford et al., 2021) have supplied valuable priors about the 3D world, sparking a surge in research on 3D generation. Notably, models such as DreamFusion (Poole et al., 2022), Magic3D (Lin et al., 2023b), and ProlificDreamer (Wang et al., 2023d) have pioneered a line of approach to per-shape optimization (Jain et al., 2022; Melas-Kyriazi et al., 2023a; Deng et al., 2023; Mohammad Khalid et al., 2022; Lee and Chang, 2022; Metzger et al., 2023; Michel et al.,

2022; Raj et al., 2023; Seo et al., 2023; Tang et al., 2023c; Wang et al., 2023a; Xu et al., 2023a,b; Tang et al., 2023b; Chen et al., 2023b; Qian et al., 2023; Chen et al., 2023c; Yu et al., 2023a). These models optimize a 3D representation (e.g., NeRF) for each unique text or image input, utilizing the 2D prior models for gradient guidance. Although they produce impressive results, these methods are hampered by prolonged optimization times, often extending to several hours, and “multi-face issue” problems.

Moreover, beyond optimization-based methods, exemplified by Zero123 (Liu et al., 2023d), numerous recent studies have investigated the employment of pre-trained 2D diffusion models for synthesizing novel views from single images or text (Shi et al., 2023c; Liu et al., 2023e; Weng et al., 2023; Ye et al., 2023; Long et al., 2024; Shi et al., 2023a). They have introduced varied strategies to foster 3D-consistent multi-view generation. The resulting multi-view images can then serve for 3D reconstruction, utilizing either optimization-based methods (Shi et al., 2023c; Liu et al., 2023e; Long et al., 2024) or feedforward models (Liu et al., 2023b, 2024a; Li et al., 2023a).

While most existing works focus on single-image-to-3D or text-to-3D, they often hallucinate regions that are invisible in the input image, which provides users with limited control over those areas. In this chapter, we seek to broaden the input to encompass unposed sparse views and address both the 3D reconstruction and pose estimation challenges in a time-efficient way—within tens of seconds.

4.2 Method: SpaRP

Given n unposed input images $\{\mathbf{I}_i \mid i = 1, \dots, n; 1 \leq n \leq 6\}$, which illustrate a single object from arbitrary categories, we predict their relative camera poses ξ_{ij} and reconstruct the 3D model $\mathcal{M}$ of the object. As illustrated in Fig. 4.2, we first finetune a 2D diffusion model (Rombach et al., 2022) to process the unposed sparse input images (Section 4.2.1). The 2D diffusion model is responsible for jointly generating grid images for both the NOCS maps of the input views, as well as multi-view images with known camera poses. We use the predicted NOCS maps to estimate the camera poses for the input views (Section 4.2.2).

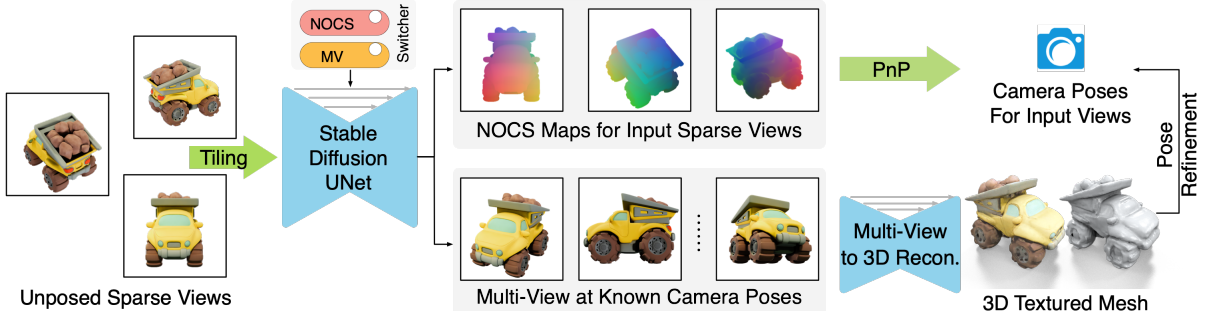


Figure 4.2: **Pipeline Overview of SpaRP.** We begin by taking a sparse set of unposed images as input, which we tile into a single composite image. This composite image is subsequently provided to the Stable Diffusion UNet to serve as the conditioning input. The 2D diffusion model is simultaneously finetuned to predict NOCS maps for the input sparse views and multi-view images under known camera poses. From the NOCS maps, we extract the camera poses corresponding to the input views. The multi-view images are then processed by a reconstruction module to generate textured 3D meshes. Optionally, the camera poses can be further refined using the generated mesh for improved accuracy.

The resulting multi-view images are fed into a two-stage 3D diffusion model for a coarse-to-fine generation of a 3D textured mesh (Section 4.2.3). This joint training strategy allows the two branches to complement each other. It enhances the understanding of both the input sparse views and the intrinsic properties of the 3D objects, thereby improving the performance of both pose estimation and 3D reconstruction. Optionally, the generated 3D mesh can also be used to further refine the camera poses (Section 4.2.4).

4.2.1 Tiling Sparse View Images as Input Condition

Recently, numerous studies have shown that 2D diffusion models not only possess robust open-world capabilities but also learn rich 3D geometric priors. For instance, Stable Diffusion (Rombach et al., 2022), can be finetuned to include camera view control (Liu et al., 2023d; Shi et al., 2023c,a; Li et al., 2023a; Liu et al., 2023e; Long et al., 2024), enabling it to predict novel views of objects—a task that necessitates significant 3D spatial reasoning. Consequently, we are inspired to utilize the rich priors inherent in 2D diffusion models for the tasks of sparse-view 3D reconstruction and pose estimation.

Unlike most existing approaches that use a single RGB image as the condition and focus on synthesizing multi-view images, our goal is to take a sparse set of input images

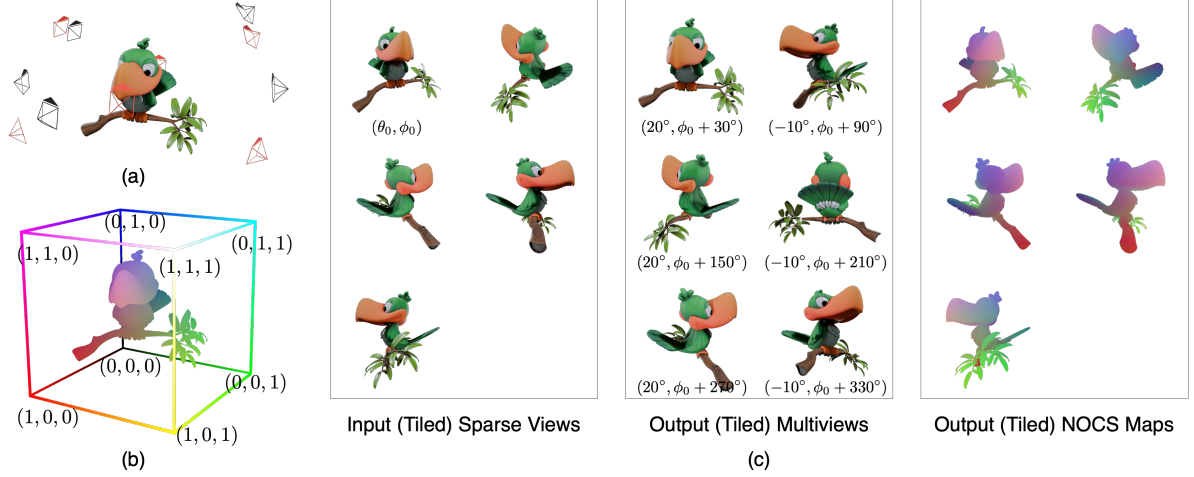


Figure 4.3: (a) Regardless of the poses of the sparse input views (in black), the output multiviews are uniformly distributed (in red) and encompass the entire 3D object. (b) The Normalized Object Coordinate Space (NOCS) of the object, whose orientation is aligned with the azimuth of the first input view. (c) An example of input and output tiled images. The elevation and azimuth of the first input view are denoted by θ_0 and ϕ_0 , respectively. The camera poses of the output multiview images are determined by ϕ_0 . The output NOCS maps correspond to the input sparse views, and the orientation of the coordinate frame is also determined by ϕ_0 .

and stimulate Stable Diffusion to infer the spatial relationships among the input views implicitly. To accomplish this, given $1 \sim 6$ sparse views from arbitrary camera poses, we tile them into a 3×2 multi-view grid, as illustrated in Fig. 4.3 (c). The image in the first grid cell determines a canonical frame (to be discussed later), while the order of the other views is inconsequential. When there are fewer than 6 sparse views, we use empty padding for the remaining grid cells. This composite image then serves as the condition for Stable Diffusion, which is expected to assimilate all information from the input sparse views during the diffusion process.

We employ Stable Diffusion 2.1 as our base model. To adapt the original text-conditioning to our tiled multi-view image condition, we follow (Shi et al., 2023a) to apply both local and global conditioning strategies. For local conditioning, we use the reference-only attention mechanism (Zhang, 2023), where we process the reference tiled image with the denoising UNet model and append the attention keys and values from this image to corresponding layers in the denoising model for the target images. This

mechanism facilitates implicit yet effective interactions between the diffusion model and the sparse views. For global conditioning, we integrate the mean-pooled CLIP embedding of all input images—modulated by learnable token weights—into the diffusion process, enhancing the model’s ability to grasp the overarching semantics and structure of the sparse views.

As depicted in Figs. 4.2 and 4.3, our objective is to concurrently generate grid images for both NOCS maps of the input views and multi-view images from known camera poses. To achieve this, we utilize a domain switcher (Long et al., 2024) that enables flexible toggling between the two domains. The switcher consists of two learnable embeddings, one for each domain, which are then injected into the UNet of the stable diffusion models by being added to its time embedding.

4.2.2 Image-to-NOCS Diffusion as a Pose Estimator

Conventional Structure-from-Motion (SfM) solvers, such as COLMAP (Schonberger and Frahm, 2016), rely on feature matching for pose estimation. However, in scenarios with sparse views, there may be little to no overlap between input views. The lack of sufficient visual correspondence cues often renders the solvers unreliable and prone to failure. Consequently, instead of relying on local correspondences, we leverage the rich semantic priors embedded in 2D diffusion models for pose estimation.

One of the primary challenges is to enable 2D diffusion models to output camera poses. While camera poses can be represented in various scalar formats (e.g., 6-dimensional vector, four-by-four matrix, etc.), they are not native representations for a 2D diffusion model to generate. Inspired by recent works demonstrating that 2D diffusion models can be used to predict normal maps (Long et al., 2024)—a domain different from natural images—we propose using a surrogate representation: the Normalized Object Coordinate Space (NOCS) (Wang et al., 2019). We finetune Stable Diffusion to predict NOCS maps for each input view.

As depicted in Fig. 4.3(b), a NOCS frame is determined for each set of input sparse

view images and the underlying 3D object. Specifically, the 3D shape is normalized into a unit cube, i.e., $x, y, z \in [0, 1]$. The shape’s upward axis aligns with the dataset’s inherent upward axis of the 3D object, typically the gravity axis. Predicting the object’s forward-facing direction may be ambiguous, so we rotate the 3D shape in the NOCS frame to align its forward direction (zero azimuth) with that of the first input view, thus unambiguously establishing the NOCS frame. For each input view, we then render a NOCS map, where each 2D pixel (r, g, b) represents the corresponding 3D point’s position (x, y, z) in the defined NOCS frame, as shown in Fig. 4.3(c). These NOCS maps align with the operational domain of 2D diffusion models, similar to the normal maps in previous work (Long et al., 2024).

To facilitate interactions between NOCS maps from different views and generate more 3D-consistent NOCS maps, we tile all NOCS maps into a 3×2 grid image as the input condition (see Section 4.2.1), following the same tiling order and the empty padding convention. We finetune Stable Diffusion to generate these multi-view tiled NOCS maps, so the 2D diffusion model can attend to both the input sparse views and their NOCS maps during the diffusion process.

After generating the NOCS maps for the input sparse views, we employ a traditional Perspective-n-Point (PnP) solver (Bradski, 2000) to compute the poses $\{\xi_i^{\text{pnp}}\}$ from the NOCS frame to the camera frames of each input view by minimizing the reprojection error:

$$\xi_i^{\text{pnp}} = \arg \min_{\xi_i \in \text{SE}(3)} \sum_{j=1}^{m_i} \|\mathbf{p}_{i,j} - \text{proj}(\mathbf{q}_{i,j}, \xi_i)\|_2^2, \quad (4.1)$$

where $\mathbf{p}_{i,j}$ represents the j^{th} pixel’s location in the i^{th} NOCS map; $\mathbf{q}_{i,j}$ is the corresponding 3D point location in the NOCS frame; m_i is the number of pixels for the i^{th} view, and proj is the perspective projection operation. Note that the PnP algorithm assumes known camera intrinsics and optimizes only for the camera extrinsics. A RANSAC scheme is applied during the PnP computation for outlier removal, enhancing the robustness of the pose prediction to boundary noises and errors from the 2D diffusion model. As all NOCS maps share a common NOCS frame, we can thus determine the relative camera poses

between views i and i' through $\xi_i^{-1}\xi_{i'}$.

4.2.3 Multi-View Prediction for 3D Reconstruction

We follow the paradigm of recent single-image-to-3D methods (Liu et al., 2024a; Li et al., 2023a) by initially generating multi-view images and subsequently using a feed-forward 3D reconstruction module to convert these images into a 3D representation. It is noteworthy that the input sparse views might not encompass the entire 3D objects, nor provide adequate information for 3D reconstruction. Therefore, we propose to predict multi-view images at uniformly distributed camera poses first, and then use these predicted images for 3D reconstruction.

Unlike traditional novel view synthesis (Liu et al., 2023d), our approach employs a fixed camera configuration for target multi-views. As depicted in Fig. 4.3, our target multi-view images consist of six views with alternating 20° and -10° elevations, and 60° -spaced azimuths relative to the first input view. Although the elevation angles are set absolutely, the azimuth angles are relative to the azimuth of the first input sparse view to resolve the ambiguity in face-forwarding directions. Furthermore, We maintain consistent camera intrinsics across target views, independent of input views. These strategies mitigate challenges in predicting camera intrinsics and elevation during the 3D reconstruction process. Existing methods hindered by this issue may be sensitive to intrinsic variations and often depend on predicting (Liu et al., 2023b) or requiring user-specified (Liu et al., 2023e; Tang et al., 2023b, 2024) input image elevations.

Similar to NOCS map prediction, we tile all six views into a 3×2 grid image and finetune Stable Diffusion to generate this tiled image. The 2D diffusion model, conditioned on the input sparse views, aims to incorporate all information from input views, deduce the underlying 3D objects, and predict the multi-view images at the predetermined camera poses. Although the predicted poses of input sparse views (Section 4.2.2) are not directly employed in the 3D reconstruction, the joint training of NOCS prediction and multi-view prediction branches implicitly complement each other and boost the performance of both

tasks.

Upon generating the multi-view images at known camera poses, we utilize the multi-view to 3D reconstruction module proposed in (Liu et al., 2024a) to lift these images to 3D. The reconstruction module adopts a two-stage coarse-to-fine approach, which involves initially extracting the 2D features of the generated multi-view images, aggregating them with the known camera poses, and constructing a 3D cost volume. This 3D cost volume acts as the condition for the 3D diffusion networks. In the coarse stage, a low-resolution 64^3 3D occupancy volume is produced. This is subsequently refined to yield a high-resolution 128^3 SDF (Signed Distance Field) volume with colors. Finally, a textured mesh is derived from the SDF volume employing the marching cubes algorithm.

4.2.4 Pose Refinement with Reconstructed 3D Model

In Section 4.2.2, we finetune diffusion models for NOCS map prediction and camera pose estimation. However, due to the hallucinatory and stochastic nature of diffusion models, unavoidable errors may exist. The generated 3D mesh $\mathcal{M}$, though not perfect, provides a multi-view consistent and explicit 3D structure. We can further refine the coarse poses predicted from the NOCS maps by leveraging the reconstructed 3D shape.

Pose Refinement via Differentiable Rendering. Starting with initial poses $\{\xi_i^{\text{pnp}}\}$ extracted from the predicted NOCS maps, we refine them through differentiable rendering (Laine et al., 2020). Specifically, we render the generated mesh $\mathcal{M}$ at optimizing camera poses ξ_i . We minimize the rendering loss between the rendered image $\mathbf{I}_i^r = \mathcal{R}(\mathcal{M}, \xi_i)$ and the input image $\mathbf{I}_i$ to obtain the optimally fitted camera pose ξ_i^* . The optimization process can be formulated as:

$$\xi_i^* = \arg \min_{\xi_i \in \text{SE}(3)} (\lambda \cdot \mathcal{L}_{\text{mask}}(\mathbf{I}_i^r, \mathbf{I}_i) + \mu \cdot \mathcal{L}_{\text{rgb}}(\mathbf{I}_i^r, \mathbf{I}_i)), \quad (4.2)$$

where $\mathcal{L}_{\text{mask}}$ and $\mathcal{L}_{\text{rgb}}$ are the cross-entropy and MSE losses computed for the foreground masks and the RGB values, respectively, and λ and μ are two weighting coefficients. The

refinement process is lightweight and can be completed in just one second, given the generated mesh.

Mixture of Experts (MoE). The NOCS pose predictions are inherently stochastic and may not produce an accurate pose in a single pass. For instance, with objects possessing certain symmetries, the diffusion model may predict only one of the possible symmetric poses. We employ a Mixture of Experts (MoE) strategy to further refine the pose, which is simple but effective. Specifically, we generate multiple NOCS maps for each input view using different seeds. We then select the pose that minimizes the rendering loss based on the refinement results with the generated 3D mesh. This technique effectively reduces pose estimation error, as quantitatively validated in Section 4.3.3.

4.3 Experiments

4.3.1 Evaluation Settings

Training Datasets and Details. We train our models on a curated subset of 100k shapes from the Objaverse dataset (Deitke et al., 2023b). Considering the variable quality of the original Objaverse dataset, we opted to filter out higher-quality data by initially manually annotating 8,000 3D objects based on overall geometry quality and texture preferences. Subsequently, we train MLP models for quality rating classification and texture score regression, utilizing their multimodal features (Liu et al., 2023a). Based on the predictions of these models, we select shapes that are rated as high-quality and have top texture scores.

For each 3D shape, we render 10 sets of images using BlenderProc (Denninger et al., 2019). Each set comprises 6 input images, 6 output multi-view images, and 6 NOCS maps. To mimic real-world conditions and ensure model robustness, we randomly sample camera intrinsics and extrinsics, as well as environment maps for the input images. For the output multi-view images, their intrinsics remain constant, while the extrinsics are derived from a fixed delta pose and the azimuth of the input images. Each set of input

Table 4.1: **Evaluation Results for Pose Estimation.** We compare our method with RelPose++ (Lin et al., 2023a), FORGE (Jiang et al., 2022), and iFusion (Wu et al., 2023a) on three unseen datasets: OmniObject3D (Wu et al., 2023c), GSO (Downs et al., 2022), and ABO (Collins et al., 2022). 500 objects are sampled for each dataset.

Dataset	Method	Rot. Err↓	Acc.@15° ↑	Acc.@30° ↑	Trans. Err↓	Time↓
GSO	RelPose++	103.24	0.011	0.033	4.84	3.6s
	FORGE	111.40	0.004	0.020	4.21	440s
	iFusion ($n_{init}=1$)	95.15	0.208	0.258	3.65	64s
	iFusion ($n_{init}=4$)	8.61	0.651	0.759	0.49	256s
	Ours (w/o refine)	13.02	0.537	0.616	0.58	10s
	Ours ($n_{init}=1$)	9.87	0.563	0.617	0.42	27s
	Ours ($n_{init}=4$)	5.28	0.750	0.787	0.23	57s
OO3D	RelPose++	105.05	0.008	0.046	7.38	3.6s
	FORGE	99.27	0.014	0.063	7.27	440s
	iFusion ($n_{init}=1$)	91.15	0.166	0.271	4.77	64s
	iFusion ($n_{init}=4$)	15.08	0.498	0.721	1.12	256s
	Ours (w/o refine)	14.75	0.508	0.725	0.90	10s
	Ours ($n_{init}=1$)	13.40	0.544	0.730	0.89	27s
	Ours ($n_{init}=4$)	10.07	0.668	0.849	0.63	57s
ABO	RelPose++	103.14	0.017	0.039	5.01	3.6s
	FORGE	110.64	0.005	0.023	4.18	440s
	iFusion ($n_{init}=1$)	96.65	0.186	0.219	3.88	64s
	iFusion ($n_{init}=4$)	8.55	0.578	0.631	0.68	256s
	Ours (w/o refine)	10.87	0.554	0.597	0.49	10s
	Ours ($n_{init}=1$)	9.30	0.565	0.600	0.43	27s
	Ours ($n_{init}=4$)	5.80	0.675	0.701	0.27	57s

and output images shares the same environment map. During training, we randomly selected between 1 and 6 views as sparse input views, with the first view of each set always included. We train the model utilizing 8 A100 GPUs for approximately 3 days.

Baselines. For 3D reconstruction, we compare our method with both state-of-the-art single-image-to-3D and sparse-view-to-3D baselines. Single-view-to-3D methods we evaluate include optimization-based approaches, such as Zero123 XL (Deitke et al., 2023a), SyncDreamer (Liu et al., 2023e), and DreamGaussian (Tang et al., 2023b), as well as feed-forward methods like One-2-3-45 (Liu et al., 2023b) and Shap-E (Jun and Nichol, 2023). For sparse-view methods, we consider two recent open-source approaches as baselines: iFusion (Wu et al., 2023a) and EscherNet (Kong et al., 2024). We utilize ThreeStudio (Guo et al., 2023)’s implementation for Zero123 XL and the official implementations for the

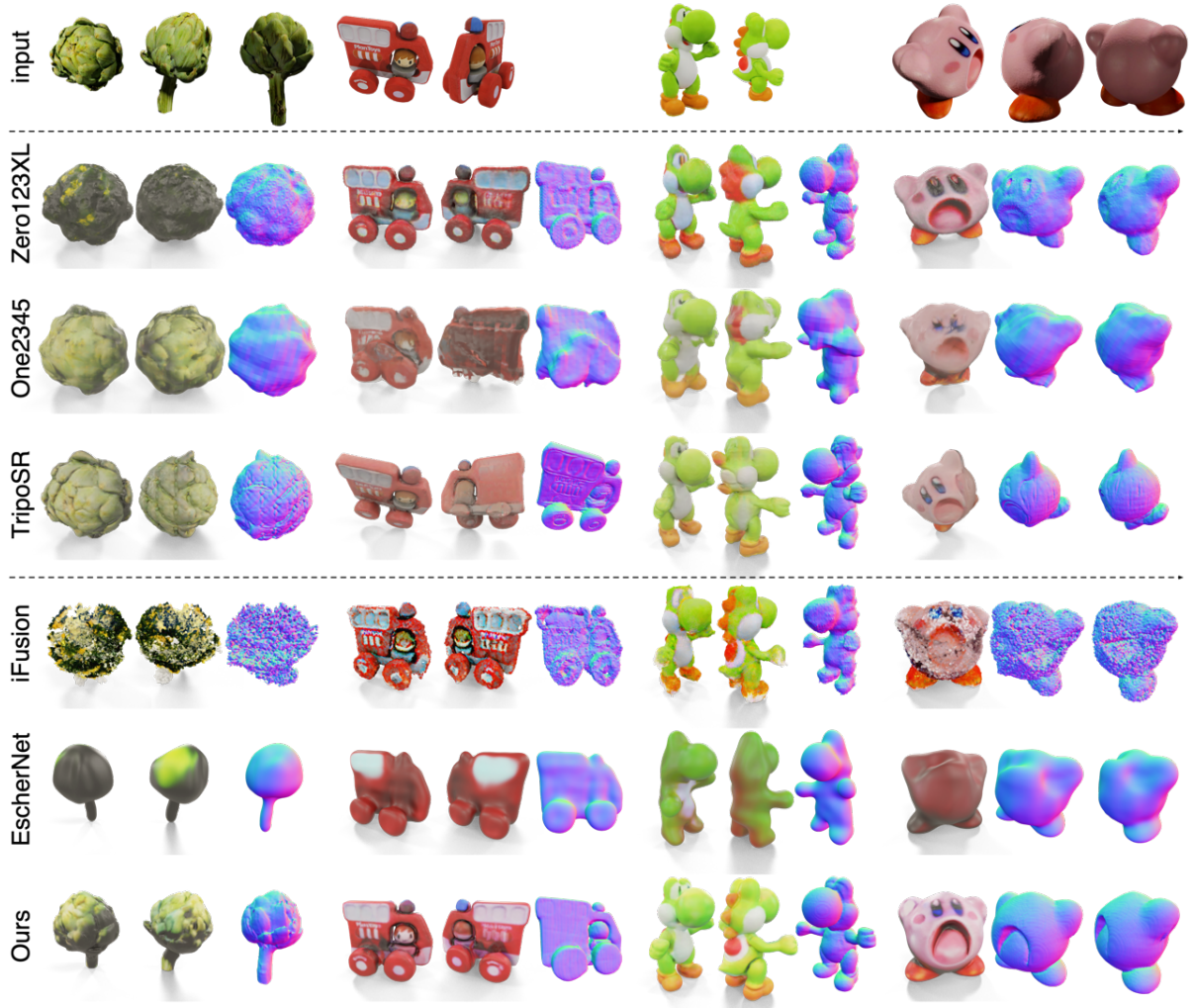


Figure 4.4: **Qualitative Results on 3D Reconstruction.** Zero123XL (Deitke et al., 2023a), One2345 (Liu et al., 2023b), and TripoSR (Tochilkin et al., 2024) are single-image-to-3D methods, each utilizing only the first input image. iFusion (Wu et al., 2023a), EscherNet (Kong et al., 2024), and our approach take all input images (the first row). Textured meshes and mesh normal renderings are shown. Shapes come from the OmniObject3D (Wu et al., 2023c) and GSO (Downs et al., 2022) datasets.

other baselines. Specifically, for iFusion, we use their official reconstruction pipeline integrated with Zero123 XL.

For sparse-view pose estimation, we compare our method with state-of-the-art approaches including RelPose++ (Lin et al., 2023a), FORGE (Jiang et al., 2022), and iFusion (Wu et al., 2023a). The latter two are optimization-based while (Lin et al., 2023a) is a feed-forward method.

Table 4.2: **Quantitative Comparison on 3D Reconstruction.** Evaluated on the complete GSO (Downs et al., 2022) dataset, which contains 1,030 3D objects. Five single-image-to-3D methods (Deitke et al., 2023a; Jun and Nichol, 2023; Liu et al., 2023b,e; Tang et al., 2023b) and two sparse-view methods (Wu et al., 2023a; Kong et al., 2024) are compared.

n_{input}	Method	F-Score (%) $\uparrow$	CLIP-Sim $\uparrow$	PSNR $\uparrow$	LPIPS $\downarrow$	Time $\downarrow$
1	Zero123 XL (Deitke et al., 2023a)	91.6	73.1	18.16	0.136	20min
	Shap-E (Jun and Nichol, 2023)	91.8	73.1	18.96	0.140	27s
	One-2-3-45 (Liu et al., 2023b)	90.4	70.8	19.07	0.133	45s
	SyncDreamer (Liu et al., 2023e)	84.8	68.9	16.86	0.145	6min
	DreamGaussian (Tang et al., 2023b)	81.0	68.4	17.88	0.147	2min
	Ours	95.7	78.2	19.87	0.124	16s
6	iFusion (Wu et al., 2023a)	88.5	66.7	16.2	0.151	28min
	EscherNet (Kong et al., 2024)	94.8	65.9	16.6	0.139	9min
	Ours	96.9	78.1	19.3	0.123	16s

Evaluation Datasets. For 3D reconstruction, we evaluate the methods on the entire GSO (Downs et al., 2022) dataset, which comprises 1,030 3D shapes; none of these shapes were seen during our training. For each 3D shape, we randomly render six views as input images. For single-image-to-3D methods, a fixed-view image is taken as input following (Liu et al., 2023b). We carefully align the predictions with the ground truth meshes before calculating the metrics.

For pose estimation, we evaluate the approaches on three datasets: OmniObject3D (Wu et al., 2023c) and GSO (Downs et al., 2022), both captured from real scans, and ABO (Collins et al., 2022), a synthetic dataset created by artists. For each dataset, we randomly choose 500 objects and render five random sparse views per shape. We follow iFusion (Wu et al., 2023a) to report the rotation accuracy and the median error in rotation and translation across all image pairs.

4.3.2 Experiment Results

Pose Prediction. We report the pose estimation results in Table 4.1, where it is evident that SpaRP outperforms all baseline methods by a significant margin. It is worth noting that RelPose++ (Lin et al., 2023a) and FORGE (Jiang et al., 2022) struggle to yield

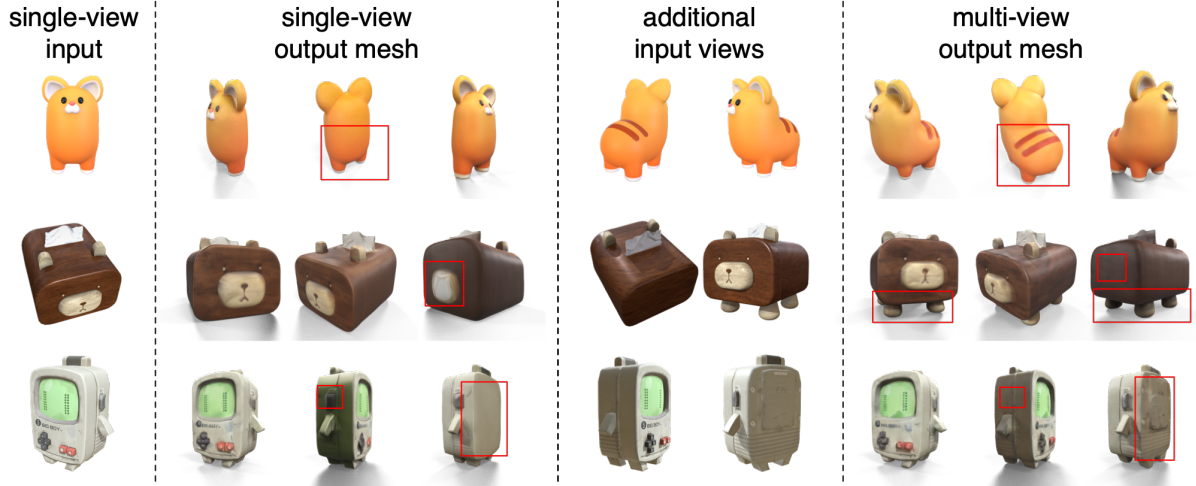


Figure 4.5: **Single-View vs. Sparse-View for 3D Reconstruction.** We compare the results of our method when using single-view and sparse-view inputs.

Table 4.3: **Ablation Study on the Number of Input Views.** Evaluated on the GSO dataset (Downs et al., 2022).

n views	Rot. Err↓	Acc.@5° ↑	Trans. Err↓	F-Score (%)↑	CLIP-Sim↑	PSNR↑
1	—	—	—	89.1	74.9	17.7
2	8.56	0.32	0.42	93.3	76.5	18.3
4	6.03	0.43	0.28	96.0	77.6	19.0
6	5.28	0.48	0.25	96.9	78.1	19.3

satisfactory results for our open-world evaluation images. iFusion, an optimization-based approach, is prone to becoming trapped in local minima. With only one initial pose ($n_{init} = 1$), it also fails to produce adequate results. In contrast, our method leverages priors from 2D diffusion models and can generate acceptable results in a single forward pass. Even without any additional refinement (w/o refine), our method can already produce results similar to iFusion with four initial poses ($n_{init} = 4$), while being far more efficient, requiring just 1/25 of the runtime. With the integration of further refinement through a mixture of experts, our method achieves even better performance.

3D Reconstruction. We present the qualitative results in Fig. 4.4. With only a single-view input, single-image-to-3D methods fail to produce meshes that faithfully match the entire structure and details of the ground truth mesh. For instance, most single-view baseline methods are unable to reconstruct the stems of the artichoke, the back of the

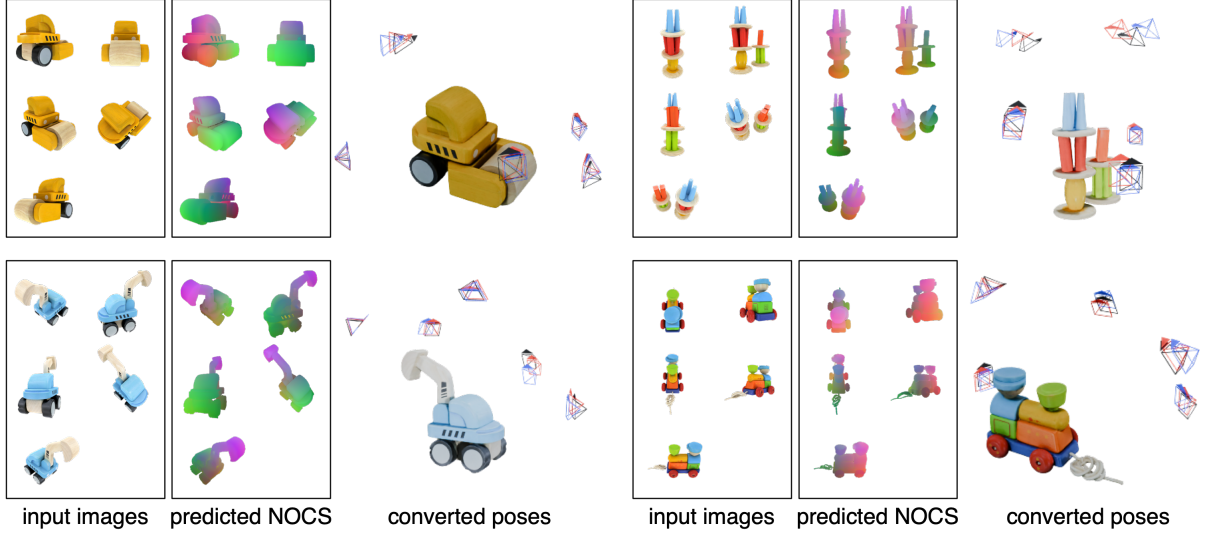


Figure 4.6: **Ablation Study on Pose Refinement.** We showcase the input images, predicted NOCS maps, and converted poses. The ground truth poses are in black, while the predicted poses before and after refinement are in blue and red, respectively.

firetruck, the red saddle on Yoshi, and the two separate legs of Kirby standing on the ground. In contrast, sparse-view methods yield results that are much closer to the ground truth by incorporating information from multiple sparse views. Compared to iFusion, EscherNet, our method generates meshes with higher-quality geometry and textures that more accurately match the input sparse views. We report the quantitative results in Table 4.2, where our method significantly outperforms both single-view-to-3D and sparse-view approaches in terms of both 2D and 3D metrics. Moreover, our method exhibits superior efficiency, being much faster than the baseline methods.

4.3.3 Analysis

Single View vs. Sparse Views. In Fig. 4.5, we present the results obtained by our method when provided with single-view and sparse-view inputs. With a single-view input, our method can still generate reasonable results, yet it may not accurately capture the structures and details of the regions that are not visible. Our method demonstrates the capability to effectively integrate information from all sparse-view inputs provided.

Number of Views. In Table 4.3, we quantitatively showcase the impact of the number of

Table 4.4: **Ablation Study on the Number of Experts for Pose Estimation.** Evaluated on the OmniObject3D (Wu et al., 2023c) and Google Scanned Objects (Downs et al., 2022) datasets.

n_{init} experts	OmniObject3D			Google Scanned Objects		
	Rot. Err ↓	Acc.@15° ↑	Trans. Err ↓	Rot. Err ↓	Acc.@15° ↑	Trans. Err ↓
1	15.23	0.495	1.04	9.83	0.562	0.43
2	12.05	0.585	0.76	6.03	0.687	0.28
4	10.46	0.647	0.68	4.71	0.805	0.21
8	9.46	0.690	0.60	4.34	0.853	0.20

Table 4.5: **Effect of Joint Training.** Evaluated on 500 objects from GSO (Downs et al., 2022).

Method	Rot. Err ↓	Acc.@15° ↑	Acc.@30° ↑	Trans. Err ↓	F-Score (%) ↑	CLIP-Sim ↑	PSNR ↑
Separate	9.87	0.563	0.617	0.42	96.81	78.36	19.03
Joint	8.57	0.601	0.677	0.37	97.12	78.90	19.42

views on both 3D reconstruction and pose estimation. We observe that incorporating more input views enables the 2D diffusion network to better grasp their spatial relationships and underlying 3D objects, boosting both tasks.

Pose Refinement. While the predicted NOCS maps can be directly converted into camera poses, we have found that these poses can be further refined through alignment with the generated 3D meshes. Fig. 4.6 showcases the predicted poses before and after refinement. Although both are generally very close to the ground truth poses, refinement can further reduce the error.

Number of Experts. We employ a mixture-of-experts strategy to address the ambiguity issues related to NOCS prediction for symmetric objects. As shown in Table 4.4, increasing the number of experts (n_{init}) from 1 to 8 led to a significant improvement in the accuracy of relative pose predictions across both the OmniObject3D (Wu et al., 2023c) and GSO (Downs et al., 2022) datasets. This demonstrates that the MoE strategy is simple yet effective in improving the robustness of our pose prediction approach.

Joint Training. We finetune 2D diffusion models to jointly predict NOCS maps and multi-view images from sparse, unposed views by leveraging a domain switcher. As shown

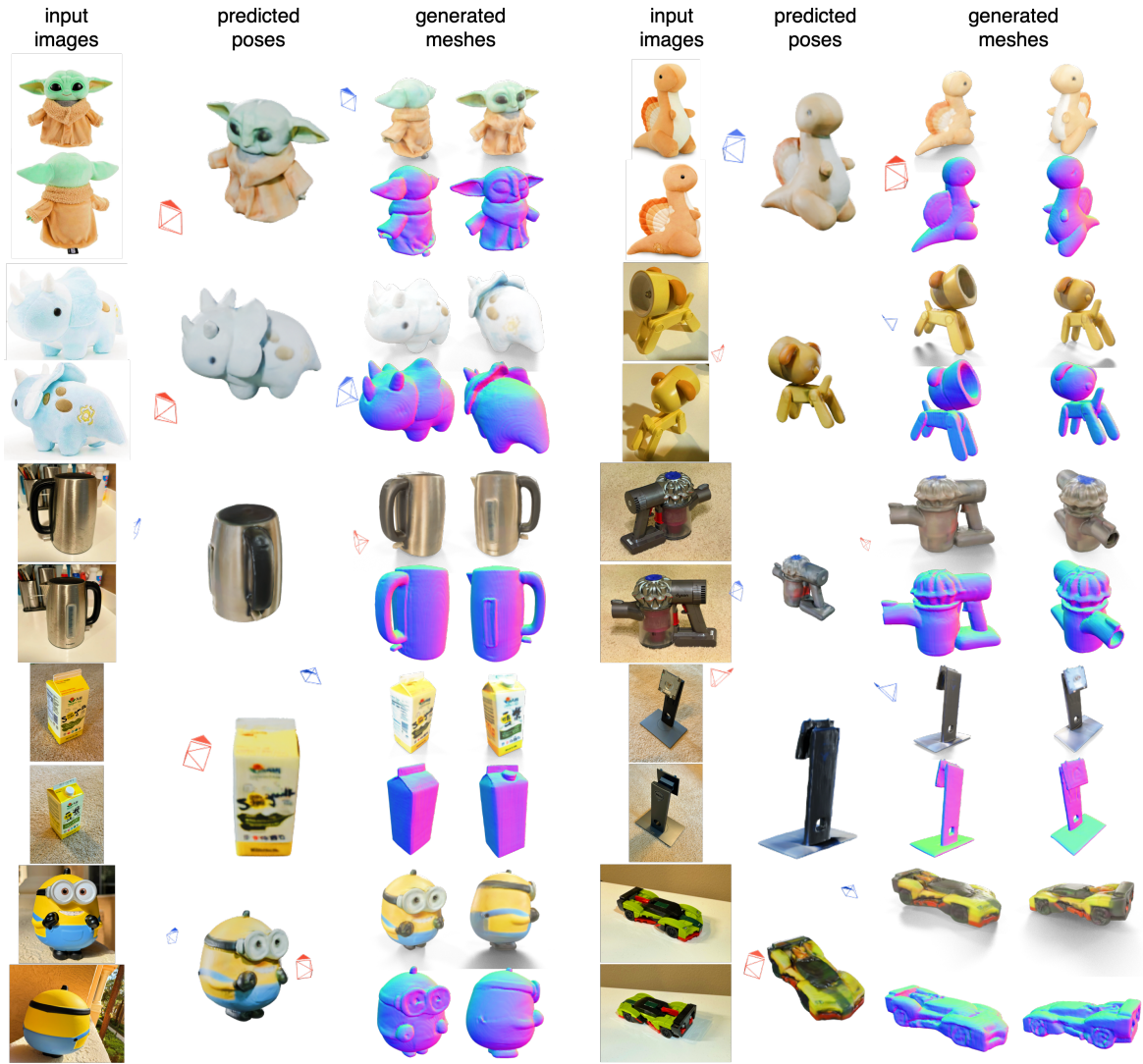


Figure 4.7: **Real-World Examples:** The input images are either sourced from amazon.com or captured using an iPhone.

in Table 4.5, this joint training strategy enables the two branches to implicitly interact and complement each other, enhancing the interpretation of both the input sparse views and the intrinsic properties of the 3D objects, which in turn improves the performance of each task.

4.4 Additional Real-World Examples

In Fig. 4.7, we demonstrate that SpaRP can be applied to real-world sparse-view images without camera poses. This includes images captured by users with consumer devices (*e.g.*, with an iPhone) or e-commerce product images (*e.g.*, from amazon.com). SpaRP is capable of achieving commendable results in both pose estimation and 3D reconstruction.

4.5 Discussion

In this chapter, we present SpaRP (Xu et al., 2024), a novel method for 3D reconstruction and pose estimation using unposed sparse-view images. Our method leverages rich priors embedded in 2D diffusion models and exhibits strong open-world generalizability. Without the need for per-shape optimization, it can deliver high-quality textured meshes, along with accurate camera poses, in approximately 20 seconds.

SpaRP intrinsically links 3D understanding (inferring spatial relationships and camera poses from sparse views) with 3D generation (reconstructing the complete object), demonstrating the synergy central to this dissertation’s approach.

While SpaRP successfully operates without known extrinsic poses, it currently assumes known camera intrinsics. Future work could aim to further enhance robustness and applicability by relaxing this assumption.

CHAPTER 5

Generalizable Part-Level 3D Understanding

The human vision system could swiftly locate the functional part upon using an object for specific tasks (Land et al., 1999). Such a critical capability in object interaction requires fine-grained object *affordance* understanding. *Affordance*, coined and originally theorized by Gibson (Gibson and Carmichael, 1966; Gibson, 1979), characterizes how humans interact with human-made objects and environments. As such, affordance understanding of objects and scenes has a significant influence on bridging visual perception and holistic scene understanding (Huang et al., 2018b,a; Chen et al., 2019) with actionable information (Soatto, 2013; Han et al., 2022).

Object affordances have two main characteristics. First, object affordances are not defined in terms of conventional categorical labels in computer vision; instead, they are defined by the associated actions for various tasks and are naturally *cross-category*. For example, both chair and sofa can be sat on, which indicates they share the *sittable* affordance. Similarly, desktop and bookshelf share the *support* affordance. Second, object affordances are intrinsically *part-based*. We could easily associate *sittable* affordance with the seats of chairs and sofas, and *support* with the boards of desktop and bookshelf. As such, the ability to learn **part-based**, **cross-category** affordance is essential to demonstrate the general object affordance understanding.

In passive affordance learning, prior literature follows the supervised learning paradigm, in which dense affordance annotation on the objects is fed as supervised signals (Deng et al., 2021). However, this line of thought depends heavily on the quality of dense annotation, which significantly deviates from how we humans learn to understand affordance. Humanlike supervision would be: “you can sit on this chair and rest your arm,” “you

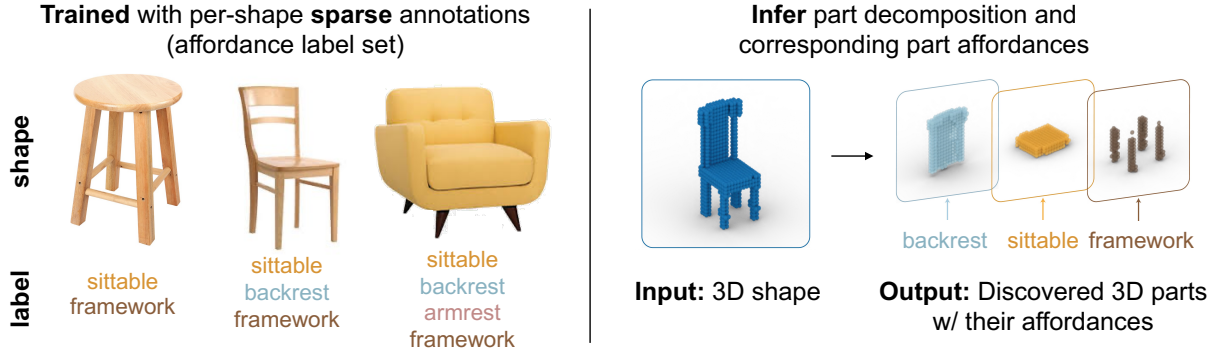


Figure 5.1: **The proposed PartAfford: discover 3D object part affordances by learning contrast in affordance compositions.** During training (**left**), given weak annotations (per-shape affordance label set), a learning framework is devised to ground affordance (*e.g.*, backrest) to 3D part (*e.g.*, sofa back) through learning cross-category, affordance-related shapes (*e.g.*, chair, sofa) with various affordance compositions. At test time (**right**), the learned model decomposes the 3D object into parts and infers the part-level affordances.

can open the lid and hold water with the cup.” In this work, we try to answer: How to distinguish each object part while recognizing corresponding affordances with such weak and natural supervision?

To tackle this problem, we present PartAfford, a new task of part-level affordance discovery, which learns the object affordance with the natural supervision of the affordance set. As shown in Fig. 5.1, by providing only the set of affordance labels for each object, the algorithm is tasked to decompose the 3D shapes into parts and discover how each part corresponds to a certain affordance category, which is challenging and under-explored in the area of generalizable part-level object understanding and affordance learning.

To address this, we propose a novel method that discovers part-level representations with self-supervised 3D reconstruction, affordance set supervision, and primitive regularization. The proposed approach consists of two main components. The first component is an encoder with slot attention for unsupervised clustering and abstraction. Specifically, we encode the 3D object into visual features and abstract the low-level features into a set of *slot* variables (Locatello et al., 2020). The second component is a decoder built upon the learned slot features. It has three output branches that jointly reconstruct the 3D parts and object, predict the affordance labels, and regularize the learned part-level

shapes with cuboidal primitives. Our method does not rely on dense supervision but instead learns from the weak set supervision. It discovers the part-level affordance by learning the correspondence between affordance labels and abstracted 3D object parts.

Learning and evaluating PartAfford demands collections of 3D objects and their affordance labels for object parts. Prior work on visual affordance learning (Hassanin et al., 2021) either focuses on 2D objects and scenes or lacks part-based annotation (Deng et al., 2021). Hence, we construct a part-level, cross-category 3D object affordance dataset annotated with 24 affordance categories shared among over 25,000 3D objects. The 3D objects are collected from PartNet dataset (Mo et al., 2019b) and the PartNet-Mobility dataset (Xiang et al., 2020). The 24 part affordance categories are defined in terms of adjectives (*e.g.*, “sittable”) or nouns (*e.g.*, “armrest”); they describe how object parts could afford human daily actions and activities. We annotate the part-level object affordances by manually mapping the fine-grained object part defined in PartNet to the part affordances defined in this work.

By experimenting on this newly constructed PartAfford dataset, we empirically demonstrate that our method jointly enables the abstraction of 3D objects and part-level affordance discovery. Our model also shows strong generalizability on hard and cross-category objects. Further experiments and ablations analyze each component’s contribution and point out future directions.

In summary, our work makes four main contributions:

- We present a new PartAfford task for part-level affordance discovery. Compared to the prior densely-supervised learning paradigm, PartAfford learns the visual object affordance more naturally.
- We propose a novel learning framework for tackling PartAfford, which jointly abstracts 3D objects into part-level representations and discovers affordances by learning the affordance correspondence.
- We build the benchmark for learning and evaluating PartAfford by curating a dataset consisting of 3D objects and annotating part-level affordances.

- We empirically demonstrate the efficacy and generalization capability of the proposed method and analyze each component’s significance via a suite of ablation studies. Code and data will be released for research purposes.

5.1 Related Work

Affordance Learning Affordance learning is a multidisciplinary research field of vision, cognition, and robotics. In general, “affordance” is first perceived from images (Gupta et al., 2011; Kjellström et al., 2011; Zhu et al., 2015; Myers et al., 2015; Roy and Todorovic, 2016) or videos (Xie et al., 2013; Zhu et al., 2016; Fang et al., 2018; Nagarajan et al., 2020), followed by cognitive reasoning (Zhu et al., 2015, 2020), and finally serves for task and motion planning in robotics (Nagarajan and Grauman, 2020; Mo et al., 2022). Prior work tackles affordance at various scales and representations. Although affordance has been studied at the scene level (Zhao and Zhu, 2013; Gupta et al., 2015; Roy and Todorovic, 2016), object level (Nguyen et al., 2017; Mo et al., 2021; Gadre et al., 2021), and associated with generated human poses (Zhu et al., 2015; Wang et al., 2017b), few attempts study affordance as a 3D shape analysis task (Yu et al., 2015; Liang et al., 2016; Zhu et al., 2016; Wang et al., 2017a) since it would normally require large-scale, high-quality 3D data. A recent work (Deng et al., 2021) benchmarks several affordance estimation tasks on PartNet (Mo et al., 2019b) with dense affordance heatmap supervisions, annotated by densely selecting keypoints without considering affordance compositionality. Prior work also tackles affordance learning through interaction-based methods, either from human demonstration videos (Kjellström et al., 2011; Nagarajan et al., 2019, 2020) or simulation-based active learning (Wang et al., 2022c; Mo et al., 2021). The first usually infers high-level and coarse 2D affordance, and the second is often restricted to basic manipulations in specific domains. In comparison, PartAfford studies affordance in a weakly supervised manner, such that the affordance discovery will be guided by affordance set matching and geometry abstraction. The new affordance dataset we construct provides fine-grained, part-level 3D affordance annotations, tailored for the weak supervision setting

and affordance compositionality study.

Object-centric Learning Object discovery has been studied in an iterative end-to-end fashion (Greff et al., 2017; Van Steenkiste et al., 2018; Burgess et al., 2019; Engelcke et al., 2019; Greff et al., 2019; Du et al., 2021). Recently, Locatello et al. (2020) presents the slot attention module, an efficient and generic framework for object-centric representation extraction. It is capable of modeling compositional nature in synthetic scenes with multiple simple geometry shapes (Kabra et al., 2019). Subsequently, Stelzner et al. (2021) and Yu et al. (2021b) apply slot attention on unsupervised 3D-aware scene decomposition, integrating NeRF (Mildenhall et al., 2020) as object representations. They demonstrate that slot-based bottleneck could perform reasonably on synthetic multi-view RGB datasets with a textureless background. In the weakly-supervised regime, methods have been proposed for 3D semantic segmentation with scene-level labels (Wei et al., 2020; Ren et al., 2021). However, these methods rely on additional input (*e.g.*, color, normal) and abstract object-level features in different data domains. Our work takes one step further to tackle the challenge of *part-level* affordance discovery of 3D objects; part discovery is more complex than object discovery, primarily due to the ambiguity in the object part segmentation without applying additional constraints. Fortunately, for man-made objects, affordances are attached to objects at the part level. This observation implies the possibility of combining part discovery and affordance learning with minimal supervision. In this work, we integrate part discovery with affordance estimation, hoping that affordance information would help discover object parts sharing similar affordances.

Unsupervised Geometric Primitive Modeling Whereas supervised geometric primitive abstraction methods (Mo et al., 2019a; Yang et al., 2020) require dense hierarchical annotations, unsupervised frameworks using cuboid-based (Tulsiani et al., 2017; Sun et al., 2019), superquadrics-based (Paschalidou et al., 2019, 2020), or other genus-zero-shape (Deng et al., 2020; Paschalidou et al., 2021) primitives discover structural information naturally embedded in the geometry. Recently, Yang and Chen (2021) unsupervisedly

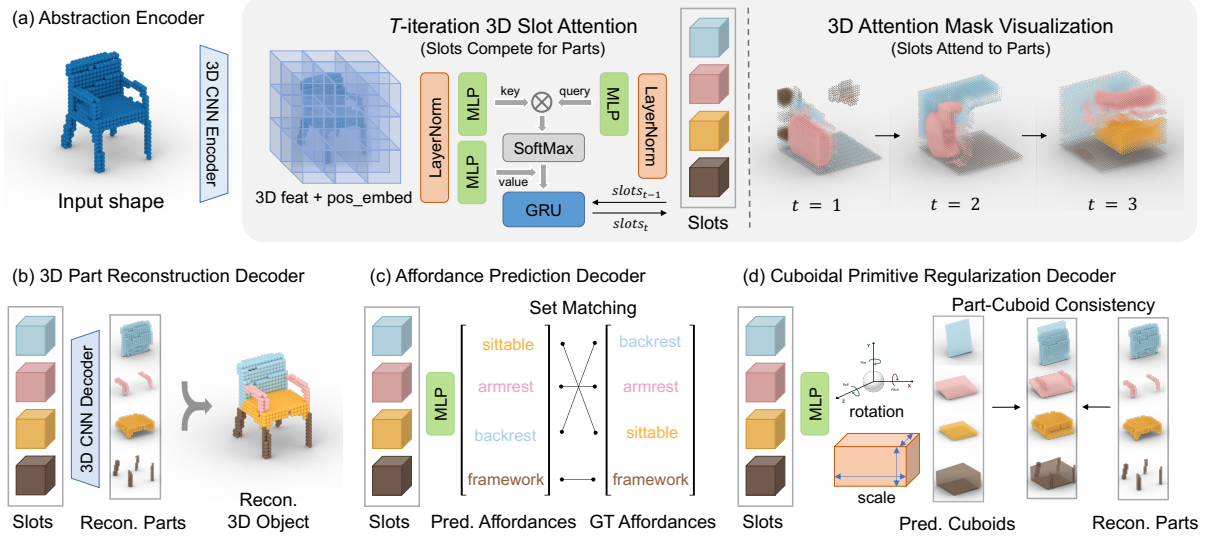


Figure 5.2: **Illustration of the proposed method for PartAfford.** Our model contains two main components: abstraction encoder and affordance decoder. (a) **Abstraction encoder** takes a 3D object as input, extracts features with 3D convolutional neural networks, and abstracts it into several slots. **Affordance decoder** with three branches jointly (b) reconstructs the 3D parts, (c) predicts affordance labels, and (d) regularizes cuboidal primitives. Each predicted cuboid in (d) wraps around the corresponding predicted object part in (b) tightly.

learn the cuboid-based shape abstraction with shape co-segmentation. Yet, it relies heavily on the ground-truth point normals for accurate abstraction and lacks semantic representation for object understanding. In our affordance discovery framework, we leverage the cuboidal regularization to refine the reconstructed affordance part, which distinguishes densely connected 3D parts by providing geometric prior, thus improving the affordance part discovery.

5.2 Task Definition

We formulate the new task PartAfford as discovering the part-level object affordance with the affordance set supervision. We define $K = 24$ common affordance categories $\mathcal{S} = \{s_k\}_{k=1}^K$, such as “sittable” and “openable,” for object understanding. Input is given as a collection of N objects $\{o_i\}_{i=1}^N$ and their corresponding affordance set labels $\{\mathcal{A}_i\}_{i=1}^N$, where $\mathcal{A}_i = \{a_i^j\}_{j=1}^{J_i}$. $a_i^j \in \mathcal{S}$, and J_i represents the number of distinct assigned affordances

for each object i . PartAfford requires an algorithm to decompose each object into parts and discover the affordance corresponding to each object part. Fig. 5.1 illustrates the PartAfford task.

5.3 Method: PartAfford

We propose a novel framework for affordance discovery from 3D objects. It integrates unsupervised part discovery with affordance set prediction and geometric primitive abstraction; see Fig. 5.2. Given a 3D shape represented by voxel grids $\mathcal{V}$ of resolution 32^3 , our method first encodes the 3D shape into visual features and abstracts it into M slots; each slot represents an abstracted high-level feature for downstream tasks. Next, we utilize a decoder with three branches to jointly (i) decode the features into parts, (ii) predict the affordance label, and (iii) regularize the parts with cuboidal primitives. Below, we describe in detail how each module is constructed and the loss design.

5.3.1 Abstraction Encoder

The encoder takes a 3D shape as input and abstracts part-centric latent codes in an unsupervised manner. It consists of a feature extraction module and a 3D slot attention module; see Fig. 5.2a.

Feature Extraction The feature embedding backbone encodes the input voxels and generates a $D = 64$ dimensional feature for each voxel. Following (Mescheder et al., 2019), voxels are encoded by five layers of 3D convolutional neural networks. The embedded feature is then augmented with absolute positional embedding (Locatello et al., 2020).

3D Slot Attention The 3D slot attention architecture, adapted from Locatello et al. (2020), serves as the part-centric representational bottleneck between the 3D feature embedding network and the downstream decoders. The encoded feature of a 3D shape is fed into an iterative attention module, where M randomly initialized slots are updated

for $T = 3$ iterations through a GRU (Cho et al., 2014). During each iteration, the attention coefficients are calculated by applying softmax normalization over the slots on the dot-product similarity between queries (*i.e.*, linearly-mapped 3D slot features) and keys (*i.e.*, linearly-mapped input features). The attention coefficients are then applied as weights for aggregating the values (*i.e.*, linearly-mapped input feature) and updating the slots.

Since the inputs-to-slots attention assignment is normalized over the 3D slots, those slots compete to attend to a clustering of the input 3D shape. Such clusterings are similar to human-defined parts on common objects. 3D slot attention masks naturally segment the object through iterations. An example of the learned 3D attention masks is shown in Fig. 5.2a.

5.3.2 Affordance Decoder

Shown in Fig. 5.2b-d, the affordance decoder takes part-centric slot features as input, followed by three branches for 3D part reconstruction, affordance prediction, and primitive regularization. The decoder parameter is shared across slots.

3D Part Reconstruction We design a 3-layer 3D transposed convolutional decoder followed by a single MLP layer to reconstruct voxel values $\hat{\mathcal{V}}^m$ and a voxel mask for each slot. The mask is normalized across slots with softmax, which generates a normalized mask $\hat{\Lambda}^m \in \mathbb{R}^{32 \times 32 \times 32}$. It is then used to compute the weighted sum of voxel values across slots and combine the reconstructed parts $\{\hat{\mathcal{V}}^m\}_{m=1}^M$ into a full 3D shape $\hat{\mathcal{V}} : \hat{\mathcal{V}} = \sum_{m=1}^M \hat{\Lambda}^m \hat{\mathcal{V}}^m$. The 3D part reconstruction branch is self-supervised by the reconstruction loss between original voxels $\mathcal{V}$ and reconstructed voxels $\hat{\mathcal{V}}$; we use the binary cross-entropy (BCE): $\mathcal{L}_{\text{recon}} = \text{BCE}(\mathcal{V}, \hat{\mathcal{V}})$.

Affordance Prediction We predict a one-hot affordance label for each slot with a two-layer MLP with sharing weights across slots for classification.

The affordance prediction branch is weakly-supervised as we do not provide affordance labels for each voxel. Instead, only the affordance label set for the entire object is used as the supervision signal. The model is tasked to learn the alignment between the abstracted parts and the affordance labels from set supervision.

As defined in Section 5.2, the ground-truth set of affordance labels for an input 3D object is denoted as $\mathcal{A}$. We denote $\hat{\mathcal{A}}$ as the set of slot affordance predictions. $\hat{\mathcal{A}}_\sigma$ is a permutation of elements in $\hat{\mathcal{A}}$, where $\sigma \in \mathfrak{G}$ and $\mathfrak{G}$ represent all $M!$ possible permutations. $\mathcal{L}_{\text{match}}$ is the pairwise matching cost between two sets, which can be calculated by mean square error (MSE) or cross-entropy:

$$\mathcal{L}_{\text{pred}} = \min_{\sigma \in \mathfrak{G}} \mathcal{L}_{\text{match}}(\mathcal{A}, \hat{\mathcal{A}}_\sigma). \quad (5.1)$$

Due to the order-invariant nature of slot modules, we apply the Hungarian matching algorithm (Kuhn, 1955), with Huber loss as the pair-wise matching cost, to calculate the set-based (Carion et al., 2020) affordance prediction loss in Eq. (5.1).

Cuboidal Primitive Regularization As a generalized soft k-means algorithm, the slot attention mechanism heavily relies on visual cues, such as the clustering of pixel colors on the image. As such, it cannot perform precisely in a crowded scene with overlapping objects even on a toy image dataset (Locatello et al., 2020). In the 3D voxel regime, segmenting objects into parts is challenging since every voxel is connected to neighboring voxels without distinguishable visual appearances.

Therefore, we introduce the cuboidal primitive regularization module, providing a geometric prior for segmentation: Human-made objects usually have geometric regularity, and cuboid is a concise structural representation for abstraction.

From each slot embedding, the cuboid abstraction module predicts a cuboid parametrized by two vectors (Yang and Chen, 2021): a scale vector $\mathbf{s} \in \mathbb{R}^3$ and a quaternion vector $\mathbf{r} \in \mathbb{R}^4$ for 3D rotation. Of note, we calculate the cuboid center from the weighted mean of voxel positions in the slot.

To evaluate how the predicted cuboid fits the reconstructed part in the m -th slot, we first compute the Euclidean distance d_i^m from each voxel $\mathbf{p}_i^m$ to its closest cuboid face. Next, we calculate the weighted sum distance for all voxels, where the weight $v_i^m \in \hat{\mathcal{V}}^m$ is the reconstructed voxel value within $[0, 1]$. Additionally, we designed a binary surface mask $f(i)$ that masks out internal voxels in the loss. We also regularize the cuboid loss by adding a cuboid scale penalty term, *i.e.*, L_1 norm for the scale vector. Thus, the loss encourages a cuboid to wrap around a solid object part tightly. The regularization loss for all the slots is defined as:

$$\mathcal{L}_{\text{cuboid}} = \sum_m \left[\lambda_{\text{scale}} \|\mathbf{s}^m\|_1 + \sum_i f(i) v_i^m d_i^m \right]. \quad (5.2)$$

Total Loss Taking together, the total training loss is the sum of 3D reconstruction loss, affordance prediction loss, and primitive regularization loss:

$$\mathcal{L}_{\text{total}} = \lambda_{\text{recon}} \mathcal{L}_{\text{recon}} + \lambda_{\text{pred}} \mathcal{L}_{\text{pred}} + \lambda_{\text{cuboid}} \mathcal{L}_{\text{cuboid}}, \quad (5.3)$$

where λ_{recon} , λ_{cuboid} , and λ_{pred} are balancing coefficients.

Of note, with the current architecture design, for the first time, we demonstrate the capability of part-level affordance discovery from set labels. The exploration of more complex and practical modules is left for future work.

5.4 Part Affordance Dataset

To benchmark PartAfford and facilitate the research in affordance understanding, we construct a part-level 3D object affordance dataset. We focus on 24 cross-category, fine-grained affordance labels as shown in Fig. 5.3. The dataset is annotated with over 25,000 3D CAD models from the PartNet dataset (Mo et al., 2019b) and 625 articulated objects among 9 categories from the PartNet-Mobility dataset in SAPIEN (Xiang et al., 2020). Below, we describe how to define part affordances and generate affordance annotation.

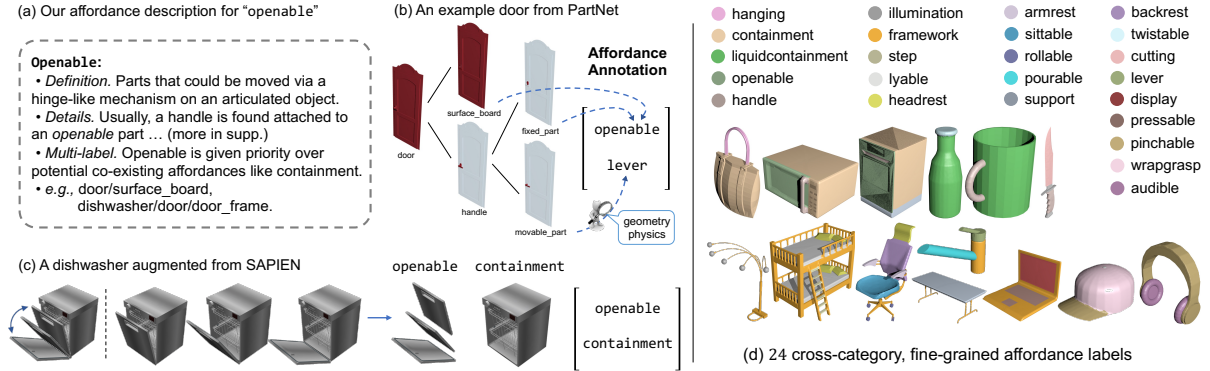


Figure 5.3: **Part affordance dataset.** (a) Description for the “openable” affordance to construct the mapping. (b) Given the part hierarchy of a door from PartNet (Mo et al., 2019b), we annotate its affordance labels by manual mapping and inspection. (c) Given a dishwasher from PartNet-Mobility (SAPIEN) (Xiang et al., 2020) and its kinematics, we rotate the door frame to include 3D objects with different articulation states. (d) Some exemplar 3D object models with color-coded affordance visualization.

Part affordances in our dataset are defined in terms of adjectives (*e.g.*, *sittable*) or nouns (*e.g.*, *armrest*), which describe how object parts could afford human daily actions and activities. We adopt certain common affordance categories from a comprehensive survey of visual affordance (Hassanin et al., 2021), *e.g.*, *containment*, *sittable*, *support*, *openable*, *rollable*, *display*, and *wrapgrasp*. However, they are coarse-grained—either at the object level or scene level. For example, “*openable*” only indicates whether an object can be opened, not on which object part can *afford* the object to be opened.

To pursue a fine-grained understanding of object affordance, we manually construct a one-to-multiple mapping from 479 kinds of object part labels defined at the finest granularity in (Mo et al., 2019b) to 24 potential affordance labels, given the detailed affordance description. We provide expert-defined descriptions for 24 affordances to guarantee the quality and consistency of the mapping construction. An example is shown in Fig. 5.3a.

Given the part hierarchy of a 3D object, we can get the corresponding affordance annotation by mapping. We also perform a manual inspection to correct the affordance labels, especially for some fine-grained parts, according to their specific geometry and physics properties. For instance, different door handles will be mapped to different

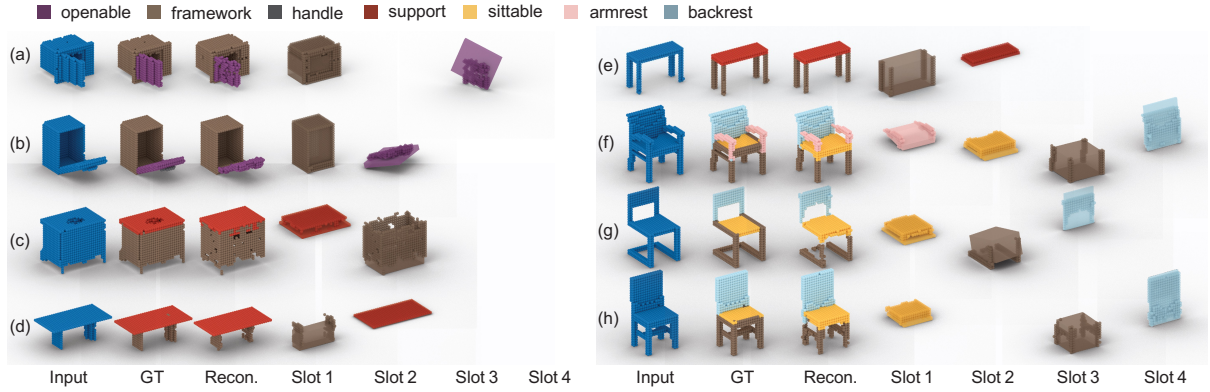


Figure 5.4: Qualitative results on three curated subsets: “openable” (a-b), “support” (c-e), and “sittable” (f-h).

affordance labels (twistable, lever, *etc.*) according to how they should be operated (Fig. 5.3b).

The PartNet dataset does not contain articulation information, making affordances such as *openable* not geometrically distinguishable. Therefore, we generate a set of shapes with *openable* affordance from the PartNet-Mobility dataset by capturing 3D shapes with various opening angles (Fig. 5.3c).

As can be seen from Fig. 5.3d, each affordance type—due to its cross-category nature—may be found on various object part instances. For example, *openable* is usually afforded by rotatable doors for unobstructed access. Under such criteria, the door frame of a dishwasher and the surface board of a door are both mapped to *openable*.

5.5 Experiments

In this section, we design and conduct comprehensive experiments to evaluate the proposed method. Fig. 5.4 visualizes our main results. We present both quantitative and qualitative comparisons of baseline models and other variants. We also evaluate the model generalization on novel shapes, analyze failure cases, and propose potential improvement directions.

5.5.1 Experimental Settings

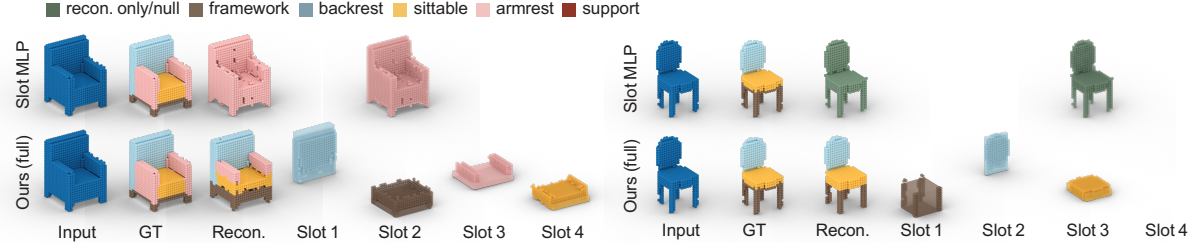
Benchmarks To benchmark PartAfford, we curate different subsets of samples from our constructed dataset. Specifically, we study the subsets related to the most representative affordance categories “sittable,” “support,” and “openable” separately, where the subsets are created by collecting all cross-category objects that have the corresponding affordance label in our dataset. They contain 7 kinds of affordances from 8 object categories, covering $17,842/25K \approx 71\%$ instances. For each subset, we learn to distinguish all the affordance labels appear in the 3D objects. Note that although a part can have multiple affordances as mentioned in Section 5.4, we only keep the most prioritized affordance for each part to ease the ambiguities in learning.

Evaluation Metrics In PartAfford, we evaluate the performances of part discovery (clustering similarity), 3D reconstruction, and affordance prediction.

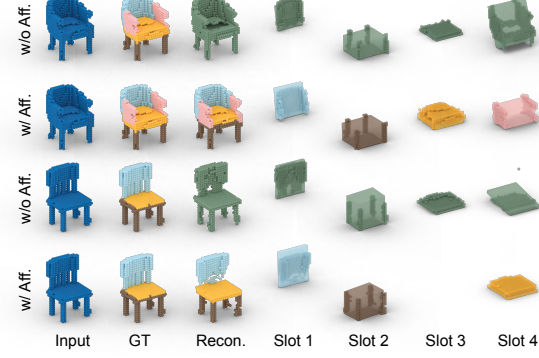
- **Part Discovery:** We use the Intersection over Union (IoU) to evaluate the part similarity. Specifically, we employ Hungarian matching to find the best matches between the reconstructed parts and ground truth parts using voxel IoU as the matching score. Then we compute the mean IoU (mIoU) by averaging the IoU between best matches.
- **3D Reconstruction:** We evaluate the shape reconstruction quality by Mean Squared Error (MSE).
- **Affordance Prediction:** Following [Locatello et al. \(2020\)](#), we use Average Precision (AP) to evaluate the prediction accuracy. A correct prediction means an exact match of the affordance label set.

Baselines and Ablations Since we are the first to propose and formulate PartAfford, we have no previous work to make direct comparisons. Therefore, we compare our method with two designed baseline models and three variants:

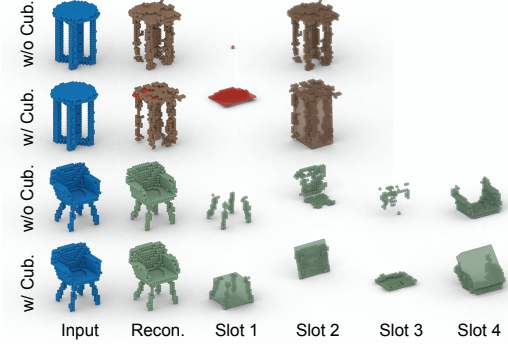
- **Slot MLP:** a simple MLP-based baseline where we replace Slot Attention with an MLP that maps from the learned feature maps (resized and flattened) to the (now ordered)



(a) Comparisons between the Slot MLP and our model.



(b) Comparisons between the models without and with affordance prediction branch. The affordance prediction facilitates the abstraction of 3D shape (rows 1-2) and elimination of spare slots (rows 3-4).



(c) Comparisons between the models without and with cuboidal primitive regularization branch. The cuboidal primitive regularization helps to guide and improve the segmentation.

Figure 5.5: Qualitative comparison results.

slot representation.

- IODINE: a baseline where we replace the Slot Attention with an object-centric learning method IODINE (Greff et al., 2019) to abstract and cluster the encoded feature.
- Ours w/o Afford & Cuboid: our model variant that only keeps the 3D part reconstruction branch.
- Ours w/o Afford: our model variant without the affordance prediction branch.
- Ours w/o Cuboid: our model variant that discards the cuboidal primitive regularization branch.
- Ours Full: our full model with all branches.

5.5.2 Implementation Details

Learning Strategy To stabilize the training, we split the training into two stages. In the first stage, we train the decoder only with 3D part reconstruction and affordance prediction branches. In the second stage, the cuboidal primitive regularization branch is incorporated into joint training with a lower learning rate.

Hyperparameter We set learning rate as 4×10^{-4} for the first stage, 2×10^{-4} for the second stage, and apply the Adam optimizer (Kingma and Ba, 2014) for optimization. It takes 5 + 9 hours on 4 RTX A6000 GPUs for two-stage full-model training of “sittable”-related objects. For slot attention, we empirically set the number of GRU iterations $T = 3$. We set the number of slots to the maximal number of affordance labels that appear in each training subset. For example, we learn the “sittable” with 4 slots, “support” with 2 slots, and “openable” with 3 slots. Section 5.6 further discusses choices of the number of slots. For the joint loss weight, we set $\lambda_{\text{recon}} = 1.0$, $\lambda_{\text{pred}} = 0.5$, $\lambda_{\text{cuboid}} = 0.1$.

5.5.3 Results and Analysis

Tables 5.1 and 5.2 tabulate the quantitative performances of all the models under different settings. Fig. 5.4 qualitatively shows the capability of our method and Fig. 5.5 compares different models. Below, we summarize some key findings:

1. The proposed method achieves the best overall performance on the PartAfford task, especially in the part discovery (mean IoU) where it outperforms the baselines by a large margin. This demonstrates the outstanding abstraction capability of our approach given the weak supervision. From Fig. 5.4, we can see our method can discover the detailed part-level representation with their aligned affordances for the 3D objects.
2. The most challenging part of PartAfford lies in the part discovery; it is also where our model differentiates from other baselines. For example, Table 5.1 shows that Slot MLP achieves the best affordance prediction performance (AP) but fails in the part discovery (mean IoU) and 3D reconstruction (MSE). As also shown in Fig. 5.5a, the

Table 5.1: **Quantitative results on “sittable.”** We evaluate the mean IoU (mIoU), mean squared error (MSE), and average precision (AP) on included objects.

Model	mIoU (%) $\uparrow$	MSE $\downarrow$	AP (%) $\uparrow$
Slot MLP	21.5	0.0150	94.5
IODINE	49.2	0.0102	92.5
Ours w/o Afford & Cuboid	31.5	0.0112	N/A
Ours w/o Afford	39.4	0.0100	N/A
Ours w/o Cuboid	55.3	0.0102	92.7
Ours (full)	57.3	0.0097	92.9

Slot MLP cannot segment the object input to parts due to the lack of abstraction capability.

- Affordance prediction branch significantly escalates the part discovery performance since it helps to learn part-affordance correspondence from the affordance composition, which provides contrasts to distinguish different parts among the training objects. Our qualitative results also show that the affordance prediction facilitates the abstraction of 3D shape (*e.g.*, rows 1-2 of Fig. 5.5b) and elimination of spare slots (*e.g.*, rows 3-4 of Fig. 5.5b);
- Cuboidal primitive regularization branch also boosts the part discovery, especially when affordance prediction is unavailable. This demonstrates that geometric priors play a crucial role in segmentation when data are not diverse enough. From Fig. 5.5c, we can see the cuboidal primitive regularization helps to segment better primitives and avoid scattered voxels.

5.5.4 Model Generalization

With the cross-category nature of affordance, we qualitatively test how the learned model can be generalized to novel objects and unseen categories. We conduct model generalization experiments by testing hard examples or objects from other categories. Examples from Fig. 5.6 demonstrate the learned model could be generalized to objects with diverse shapes. We show the results of testing the learned model on a novel object shape (bean bag) (a) from Fu et al. (2021) and unseen categories (b-d). For example, (b)

Table 5.2: **Quantitative results on “support”, “openable”, “display”, and “cutting.”**

	Model	mIoU (%) $\uparrow$	MSE $\downarrow$	AP (%) $\uparrow$
support	Slot MLP	36.8	0.0099	91.6
	Ours w/o Afford	34.8	0.0092	N/A
	Ours w/o Cuboid	51.3	0.0087	95.2
	Ours (full)	52.7	0.0085	95.1
openable	Slot MLP	21.0	0.0130	70.8
	Ours w/o Afford	19.9	0.0104	N/A
	Ours w/o Cuboid	46.7	0.0097	55.8
	Ours (full)	47.6	0.0093	60.4
display	Slot MLP	17.2	0.0163	97.2
	Ours w/o Afford & Cuboid	14.8	0.0146	N/A
	Ours w/o Cuboid	47.0	0.0140	80.6
	Ours (full)	47.6	0.0135	82.0
cutting	Slot MLP	21.6	0.0030	97.7
	Ours w/o Afford & Cuboid	27.8	0.0028	N/A
	Ours w/o Cuboid	28.1	0.0027	96.9
	Ours (full)	29.8	0.0026	97.7

shows the result of learning with “support” and testing on an “openable” object (*i.e.*, a microwave). Despite the imperfect reconstructions, partly due to the reconstruction bottleneck’s impact on disentanglement quality (Engelcke et al., 2020), the learned model can identify the functional parts given novel objects.

5.5.5 Failure Cases

We show some failure cases of our method in Fig. 5.7. For “sittable” and “support,” the failures are commonly caused by (i) the difficulties in reconstructing the fine-grained details of 3D objects with novel shapes; (ii) certain parts that violate the cuboid assumption, and thus hurt other components.

For objects in “openable” category, our model cannot discover and reconstruct “handle,” as shown in Fig. 5.4. This is because the objects with related affordances come from various object categories with diverse shapes, making it challenging for the model to capture such complex mixtures of distributions and reconstruct fine-grained 3D shapes, especially tiny parts. This points out future directions to better understand object parts

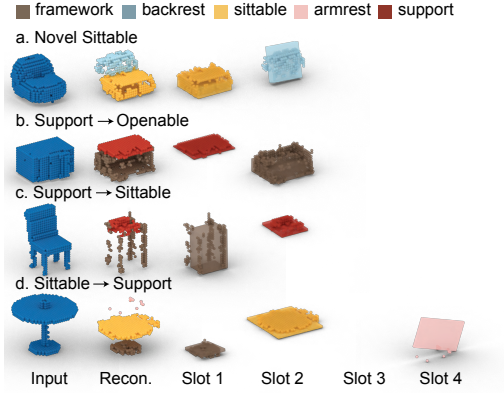


Figure 5.6: Generalization results.

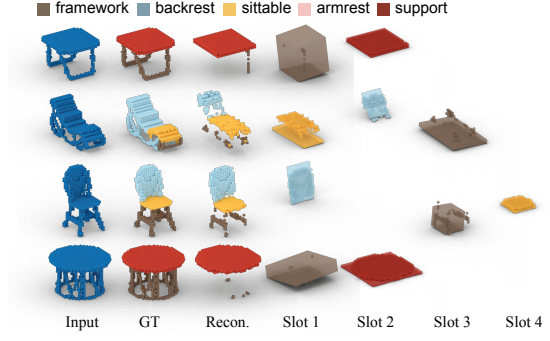


Figure 5.7: Failure cases.



Figure 5.8: Example affordance discovery model trained with $T = 3$ attention iterations. Attention is visualized in various colors and point radius. The point is colored according to the predicted affordance label of the slot which attends the most to the point. Point radius positively depends on the maximal attention value across the slots. We use trilinear upsampling to rescale the attention mask to the input resolution ($32 \times 32 \times 32$).

(*e.g.*, segment, reconstruct), and potentially an interactive learning framework to learn beyond geometry and appearance.

5.6 Further Analysis

Attention Visualization Following the settings in [Locatello et al. \(2020\)](#), we train the models using $T = 3$ attention iterations in the slot attention module. In Fig. 5.8, we observe that each slot gradually attends to the correct part of the object as the number of attention iterations increases. The attention mask at $t = 3$ depicts the silhouette of



Figure 5.9: Model performance when the number of slots increases. The model learns the “sittable” part in a “null” slot.

the reconstructed shape and parts.

Number of Slots We set the number of slots as the maximal number of affordance labels that appear in one subset, which is different from previous object-centric learning algorithm (Locatello et al., 2020), where the number of slots could be arbitrary. This is because when we increase the number of slots, we also increase the ambiguities of the affordance composition and set matching at the same time. It prevents the model from learning accurate correspondence between affordance labels and parts. As shown in Fig. 5.9, the model learns the “sittable” in a “null” slot.

Selection of Studied Affordances Although we annotate objects with 24 affordance categories, we benchmark *PartAfford* only on three subsets with 7 kinds of affordances in this work. The main reason is that we find it much more challenging to discover part-level affordance for some other affordance categories. For example, “rollable” and “cutting” usually connect to tiny object parts that are challenging to be segmented from objects, “illumination” and “display” cannot be distinguished from the objects without a deeper understanding of the visual appearance and reflections, “pressable” and “pinchable” require richer interactions to be discovered. In summary, it is either especially challenging to segment or requires more than geometric information (*e.g.*, active interactions) to discover the other affordance categories. We believe that future solutions to our task could involve those interaction cues to take the best of both worlds, further improving the performance and extending the scope. Our solution could also provide an intuitive prior for active learning methods, which could boost learning efficiency compared to learning

from scratch. We hope to further explore the learning of these affordances in the future.

Ambiguities in Affordance Learning Affordance is naturally ambiguous since the functions of object parts are rich. This work provides a well-defined benchmark to study how to learn affordance from accurate affordance definition and sparse set supervision. In another work, (Deng et al., 2021) models the multiple affordances with a mixture of distributions. Our dataset has supported learning multiple affordances per part; we annotate a single part with multiple affordances (see details in Section 5.4). Yet, as the first step towards affordance discovery, our current method focuses on learning the most important affordance. We believe our current learning framework could also be extended to learn multiple affordances by switching the one-hot affordance label to multi-hot labels. We also suggest further fine-tuning the multi-hot affordance prediction based on the one-hot version, since the affordance prediction of single-hot labels appears significant to guide the slot competition from our ablation study.

Potential Impacts for Vision & Robotics Community By learning to discover the part-level affordance, our model could facilitate the understanding of human-object interaction and object manipulation. The learned affordance could be applied to synthesize potential human actions and interactions with various 3D objects. Additionally, our work could provide priors for robotic cross-category object manipulation and task planning (Toussaint et al., 2019; Garrett et al., 2020, 2021). For example, (Hadjivelichkov et al., 2022) demonstrates a single reference image of an object with annotated affordance regions can help the robot to use the affordance skill in the real-world setting, where our method can provide generalizable affordance information on the 3D shapes without the need for manual annotation.

5.7 Discussion

In this chapter, we present PartAfford, a new task in visual affordance research that aims at discovering part-level affordances from 3D shapes. We propose a novel learning framework that discovers part-level affordances by leveraging only the affordance set supervision and geometric primitive regularization. With comprehensive experiments and analyses, we point out potential directions for incorporating visual appearance to facilitate better shape abstraction and combining it with an active learning approach for efficient affordance learning.

5.8 Related Project

The PartAfford pipeline (Xu et al., 2022), detailed in this chapter, focuses on 3D part affordance discovery. Following a similar category-agnostic principle, the related GPartNet project shifts the focus to supervised learning of articulated part actionability. Grounded in the idea that robots can manipulate geometrically similar parts in similar ways across different objects, this work specifically targets understanding and interacting with movable parts like handles and knobs. A brief overview of GPartNet is provided below.

5.8.1 GPartNet

We propose to learn such cross-category skills via Generalizable and Actionable Parts (GAParts). By identifying and defining 9 GPart classes (lids, handles, *etc.*) in 27 object categories, we construct a large-scale part-centric interactive dataset, GPartNet, where we provide rich, part-level annotations (semantics, poses) for 8,489 part instances on 1,166 objects. Based on GPartNet, we investigate three cross-category tasks: part segmentation, part pose estimation, and part-based object manipulation. Given the significant domain gaps between seen and unseen object categories, we propose a robust 3D segmentation method from the perspective of domain generalization by integrating adversarial learning techniques. Our method outperforms all existing methods by a large

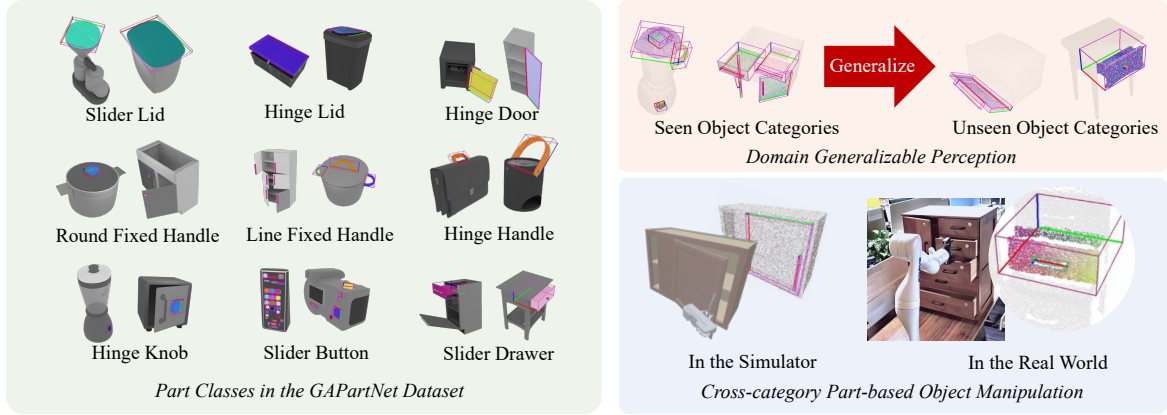


Figure 5.10: We propose to learn generalizable object perception and manipulation skills via Generalizable and Actionable Parts, and present GAPartNet, a large-scale interactive dataset with rich part annotations. We propose a domain generalization method for cross-category part segmentation and pose estimation. Our GAPart definition boosts cross-category object manipulation and can transfer to real.

margin, no matter on seen or unseen categories. Furthermore, with part segmentation and pose estimation results, we leverage the GAPart pose definition to design part-based manipulation heuristics that can generalize well to unseen object categories in both the simulator and the real world.

The overview of the system can be found in Fig. 5.10. For more details, please refer to (Geng et al., 2023).

CHAPTER 6

Conclusion

In this dissertation, we have explored methodologies to advance the generalization capabilities of 3D vision models for both generation and understanding, extending their applicability well beyond the limited object categories. We pioneered an evolving series of open-world 3D generation models, marrying large-scale 2D generative priors with 3D reconstruction techniques, thereby combating the challenges of 3D data scarcity and difficult input conditions like sparsity or unknown pose. Concurrently, we studied how to computationally identify functional similarities encoded in local geometry across diverse, category-agnostic object parts.

To recapitulate, a primary thrust involved bridging 2D potential with 3D challenges. We began with One-2-3-45, which established an efficient feed-forward paradigm generating 3D meshes from single images using 2D pretrained priors, serving as a crucial, rapid bridge between the 2D and 3D vision communities. This connection illuminated the symbiotic relationship between domains, demonstrating how 3D models extend 2D generative capabilities while 2D generalization techniques unlock new possibilities for 3D tasks, inspiring considerable subsequent work exploring this synergy. Building on this, One-2-3-45++ tackled the critical bottleneck of multi-view inconsistency by integrating joint view synthesis with robust multi-view conditioned 3D diffusion models, markedly enhancing the fidelity and quality achievable from a single image. We further pushed generalization by developing SpaRP, which uniquely addresses the joint task of shape reconstruction and pose estimation from sparse, unposed views—inherently linking geometric understanding with shape generation—demonstrating robust performance even with ambiguous real-world inputs. Complementing these generative advancements, PartAfford explored part-level

functional 3D understanding, introducing techniques to discover cross-category part affordances from geometry with weak supervision.

The influence of these efforts extends beyond the immediate vision and robotics research communities. Witnessing the broad adoption and interest these ideas have sparked—ranging from individual 3D hobbyists, game developers, and animation studios, and even finding applications in wildlife conservation—has been profoundly encouraging and highlights the tangible impact of pursuing generalizable 3D intelligence and digital twin creation. This journey, intertwining generative capabilities with functional understanding, charts a promising path toward more perceptive and interactive AI systems.

Looking ahead, our goal is to build faithful computational models of the physical world. Creating 3D assets that look visually plausible is merely a starting point. Compared with hand-crafted meshes, we still lack high-quality topology, editable structure, riggable articulation, and physically correct materials. Real-world objects push us further: interiors, functional parts, intricate joints, soft bodies, and countless other physical properties remain hard to capture. Our inability to model them exposes gaps in true understanding, aggravated by data scarcity in these niche yet vital domains. It is hoped that the explorations in this dissertation lay a cornerstone for scaling simulation and embodied AI research; overcoming the challenges above will be key milestones on the path toward physically grounded artificial general intelligence.

REFERENCES

- Achlioptas, P., Diamanti, O., Mitliagkas, I., and Guibas, L. (2018). Learning representations and generative models for 3d point clouds. In *International conference on machine learning*, pages 40–49. PMLR. 9, 29
- Aneja, S., Thies, J., Dai, A., and Nießner, M. (2022). Clipface: Text-guided editing of textured 3d morphable models. *arXiv preprint arXiv:2212.01406*. 8
- Barnes, C., Shechtman, E., Finkelstein, A., and Goldman, D. B. (2009). Patchmatch: A randomized correspondence algorithm for structural image editing. *ACM Trans. Graph.*, 28(3):24. 47
- Bradski, G. (2000). Perspective-n-point (pnp) pose computation (the opencv library). https://docs.opencv.org/4.x/d5/d1f/calib3d_solvePnP.html. 49, 57
- Burgess, C. P., Matthey, L., Watters, N., Kabra, R., Higgins, I., Botvinick, M., and Lerchner, A. (2019). Monet: Unsupervised scene decomposition and representation. *arXiv preprint arXiv:1901.11390*. 73
- Canfes, Z., Atasoy, M. F., Dirik, A., and Yanardag, P. (2023). Text and image guided 3d avatar generation and manipulation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 4421–4431. 8
- Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., and Zagoruyko, S. (2020). End-to-end object detection with transformers. In *Eur. Conf. Comput. Vis.* 77
- Chan, E. R., Nagano, K., Chan, M. A., Bergman, A. W., Park, J. J., Levy, A., Aittala, M., De Mello, S., Karras, T., and Wetzstein, G. (2023). Genvs: Generative novel view synthesis with 3d-aware diffusion models. 31, 50
- Chang, A. X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., et al. (2015). Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*. 1, 5, 9
- Chen, A., Xu, Z., Geiger, A., Yu, J., and Su, H. (2022). Tensorf: Tensorial radiance fields. In *European Conference on Computer Vision*, pages 333–350. Springer. 7, 13, 38
- Chen, A., Xu, Z., Zhao, F., Zhang, X., Xiang, F., Yu, J., and Su, H. (2021). Mvsnerf: Fast generalizable radiance field reconstruction from multi-view stereo. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14124–14133. 7, 10, 14, 15, 22, 30, 50
- Chen, D. Z., Siddiqui, Y., Lee, H.-Y., Tulyakov, S., and Nießner, M. (2023a). Text2tex: Text-driven texture synthesis via diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18558–18568. 9, 30

- Chen, R., Chen, Y., Jiao, N., and Jia, K. (2023b). Fantasia3d: Disentangling geometry and appearance for high-quality text-to-3d content creation. *arXiv preprint arXiv:2303.13873*. 30, 53
- Chen, Y., Huang, S., Yuan, T., Qi, S., Zhu, Y., and Zhu, S.-C. (2019). Holistic++ scene understanding: Single-view 3d holistic scene parsing and human pose estimation with human-object interaction and physical commonsense. In *IEEE Conf. Comput. Vis. Pattern Recog.* 69
- Chen, Z., Wang, F., and Liu, H. (2023c). Text-to-3d using gaussian splatting. *arXiv preprint arXiv:2309.16585*. 30, 53
- Cheng, Y.-C., Lee, H.-Y., Tulyakov, S., Schwing, A., and Gui, L. (2022). Sdfusion: Multimodal 3d shape completion, reconstruction, and generation. *arXiv preprint arXiv:2212.04493*. 9
- Cheng, Y.-C., Lee, H.-Y., Tulyakov, S., Schwing, A. G., and Gui, L.-Y. (2023). Sdfusion: Multimodal 3d shape completion, reconstruction, and generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4456–4465. 29
- Chiu, H.-P., Kaelbling, L. P., and Lozano-Pérez, T. (2009). Automatic class-specific 3d reconstruction from a single image. *CSAIL*, pages 1–9. 5
- Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*. 76
- Chou, G., Bahat, Y., and Heide, F. (2023). Diffusion-sdf: Conditional generative modeling of signed distance functions. In *The IEEE International Conference on Computer Vision (ICCV)*. 5
- Choy, C. B., Xu, D., Gwak, J., Chen, K., and Savarese, S. (2016). 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In *Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part VIII 14*, pages 628–644. Springer. 5, 9, 29
- Collins, J., Goel, S., Deng, K., Luthra, A., Xu, L., Gundogdu, E., Zhang, X., Vicente, T. F. Y., Dideriksen, T., Arora, H., et al. (2022). Abo: Dataset and benchmarks for real-world 3d object understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21126–21136. 61, 63
- Deitke, M., Liu, R., Wallingford, M., Ngo, H., Michel, O., Kusupati, A., Fan, A., Laforte, C., Voleti, V., Gadre, S. Y., et al. (2023a). Objaverse-xl: A universe of 10m+ 3d objects. *arXiv preprint arXiv:2307.05663*. 38, 42, 43, 44, 61, 62, 63
- Deitke, M., Schwenk, D., Salvador, J., Weihs, L., Michel, O., VanderBilt, E., Schmidt, L., Ehsani, K., Kembhavi, A., and Farhadi, A. (2022). Objaverse: A universe of annotated 3d objects. *arXiv preprint arXiv:2212.08051*. 12, 14, 17, 19, 20

- Deitke, M., Schwenk, D., Salvador, J., Weihs, L., Michel, O., VanderBilt, E., Schmidt, L., Ehsani, K., Kembhavi, A., and Farhadi, A. (2023b). Objaverse: A universe of annotated 3d objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13142–13153. 34, 37, 47, 50, 60
- Deng, B., Genova, K., Yazdani, S., Bouaziz, S., Hinton, G., and Tagliasacchi, A. (2020). Cvxnet: Learnable convex decomposition. In *IEEE Conf. Comput. Vis. Pattern Recog.* 73
- Deng, C., Jiang, C., Qi, C. R., Yan, X., Zhou, Y., Guibas, L., Anguelov, D., et al. (2022). Nerdi: Single-view nerf synthesis with language-guided diffusion as general image priors. *arXiv preprint arXiv:2212.03267*. 9, 10
- Deng, C., Jiang, C., Qi, C. R., Yan, X., Zhou, Y., Guibas, L., Anguelov, D., et al. (2023). Nerdi: Single-view nerf synthesis with language-guided diffusion as general image priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20637–20647. 30, 52
- Deng, S., Xu, X., Wu, C., Chen, K., and Jia, K. (2021). 3d affordancenet: A benchmark for visual object affordance understanding. In *IEEE Conf. Comput. Vis. Pattern Recog.* 69, 71, 72, 88
- Denninger, M., Sundermeyer, M., Winkelbauer, D., Zidan, Y., Olefir, D., Elbadrawy, M., Lodhi, A., and Katam, H. (2019). Blenderproc. *arXiv preprint arXiv:1911.01911*. 60
- Denninger, M., Winkelbauer, D., Sundermeyer, M., Boerdijk, W., Knauer, M., Strobl, K. H., Humt, M., and Triebel, R. (2023). Blenderproc2: A procedural pipeline for photorealistic rendering. *Journal of Open Source Software*, 8(82):4901. 17
- Downs, L., Francis, A., Koenig, N., Kinman, B., Hickman, R., Reymann, K., McHugh, T. B., and Vanhoucke, V. (2022). Google scanned objects: A high-quality dataset of 3d scanned household items. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 2553–2560. IEEE. 19, 20, 38, 39, 42, 43, 44, 61, 62, 63, 64, 66
- Du, Y., Li, S., Sharma, Y., Tenenbaum, J., and Mordatch, I. (2021). Unsupervised learning of compositional energy concepts. *Advances in Neural Information Processing Systems (NeurIPS)*. 73
- Engelcke, M., Jones, O. P., and Posner, I. (2020). Reconstruction bottlenecks in object-centric generative models. *arXiv preprint arXiv:2007.06245*. 85
- Engelcke, M., Kosiorek, A. R., Jones, O. P., and Posner, I. (2019). Genesis: Generative scene inference and sampling with object-centric latent representations. *arXiv preprint arXiv:1907.13052*. 73

- Erkoç, Z., Ma, F., Shan, Q., Nießner, M., and Dai, A. (2023). Hyperdiffusion: Generating implicit neural fields with weight-space diffusion. *arXiv preprint arXiv:2303.17015*. 29
- Fan, H., Su, H., and Guibas, L. J. (2017). A point set generation network for 3d object reconstruction from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 605–613. 5, 9, 29
- Fang, K., Wu, T.-L., Yang, D., Savarese, S., and Lim, J. J. (2018). Demo2vec: Reasoning object affordances from online videos. In *IEEE Conf. Comput. Vis. Pattern Recog.* 72
- Fu, H., Jia, R., Gao, L., Gong, M., Zhao, B., Maybank, S., and Tao, D. (2021). 3d-future: 3d furniture shape with texture. *Int. J. Comput. Vis.*, 129(12):3313–3337. 84
- Gadre, S. Y., Ehsani, K., and Song, S. (2021). Act the part: Learning interaction strategies for articulated object part discovery. In *Int. Conf. Comput. Vis.*, pages 15752–15761. 72
- Gal, R., Alaluf, Y., Atzmon, Y., Patashnik, O., Bermano, A. H., Chechik, G., and Cohen-Or, D. (2022). An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618*. 10
- Gao, J., Shen, T., Wang, Z., Chen, W., Yin, K., Li, D., Litany, O., Gojcic, Z., and Fidler, S. (2022). Get3d: A generative model of high quality 3d textured shapes learned from images. *Advances In Neural Information Processing Systems*, 35:31841–31854. 9, 29
- Garrett, C. R., Chitnis, R., Holladay, R., Kim, B., Silver, T., Kaelbling, L. P., and Lozano-Pérez, T. (2021). Integrated task and motion planning. *Annual review of control, robotics, and autonomous systems*, 4:265–293. 88
- Garrett, C. R., Lozano-Pérez, T., and Kaelbling, L. P. (2020). Pddlstream: Integrating symbolic planners and blackbox samplers via optimistic adaptive planning. In *Proceedings of the International Conference on Automated Planning and Scheduling*, volume 30, pages 440–448. 88
- Geng, H., Xu, H., Zhao, C., Xu, C., Yi, L., Huang, S., and Wang, H. (2023). Gapartnet: Cross-category domain-generalizable object perception and manipulation via generalizable and actionable parts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7081–7091. 90
- Gibson, J. J. (1979). *The ecological approach to visual perception*. Houghton, Mifflin and Company. 69
- Gibson, J. J. and Carmichael, L. (1966). *The senses considered as perceptual systems*, volume 2. Houghton Mifflin Boston. 69

- Girdhar, R., Fouhey, D. F., Rodriguez, M., and Gupta, A. (2016). Learning a predictable and generative vector representation for objects. In *Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part VI 14*, pages 484–499. Springer. 5, 9
- Greff, K., Kaufman, R. L., Kabra, R., Watters, N., Burgess, C., Zoran, D., Matthey, L., Botvinick, M., and Lerchner, A. (2019). Multi-object representation learning with iterative variational inference. In *Int. Conf. Mach. Learn.* 73, 82
- Greff, K., Van Steenkiste, S., and Schmidhuber, J. (2017). Neural expectation maximization. *arXiv preprint arXiv:1708.03498*. 73
- Groueix, T., Fisher, M., Kim, V. G., Russell, B. C., and Aubry, M. (2018). A papier-mâché approach to learning 3d surface generation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 216–224. 9, 29
- Guo, Y.-C., Liu, Y.-T., Shao, R., Laforte, C., Voleti, V., Luo, G., Chen, C.-H., Zou, Z.-X., Wang, C., Cao, Y.-P., and Zhang, S.-H. (2023). threestudio: A unified framework for 3d content generation. <https://github.com/threestudio-project/threestudio>. 38, 40, 61
- Gupta, A., Satkin, S., Efros, A. A., and Hebert, M. (2011). From 3d scene geometry to human workspace. In *IEEE Conf. Comput. Vis. Pattern Recog.* 72
- Gupta, A., Xiong, W., Nie, Y., Jones, I., and Oğuz, B. (2023). 3dgen: Triplane latent diffusion for textured mesh generation. *arXiv preprint arXiv:2303.05371*. 9, 29
- Gupta, S., Arbeláez, P., Girshick, R., and Malik, J. (2015). Indoor scene understanding with rgb-d images: Bottom-up segmentation, object detection and semantic segmentation. *Int. J. Comput. Vis.*, 112(2):133–149. 72
- Hadjivelichkov, D., Zwane, S., Deisenroth, M., Agapito, L., and Kanoulas, D. (2022). One-shot transfer of affordance regions? affcorrs! *arXiv preprint arXiv:2209.07147*. 88
- Han, M., Zhang, Z., Jiao, Z., Xie, X., Zhu, Y., Zhu, S.-C., and Liu, H. (2022). Scene reconstruction with functional objects for robot autonomy. *Int. J. Comput. Vis.*, pages 1–22. 69
- Hassanin, M., Khan, S., and Tahtali, M. (2021). Visual affordance and function understanding: A survey. *ACM Computing Surveys (CSUR)*, 54(3):1–35. 71, 79
- Henzler, P., Reizenstein, J., Labatut, P., Shapovalov, R., Ritschel, T., Vedaldi, A., and Novotny, D. (2021). Unsupervised learning of 3d object categories from videos in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4700–4709. 10
- Ho, J., Jain, A., and Abbeel, P. (2020). Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851. 35

- Hong, F., Zhang, M., Pan, L., Cai, Z., Yang, L., and Liu, Z. (2022). Avatar-clip: Zero-shot text-driven generation and animation of 3d avatars. *arXiv preprint arXiv:2205.08535*. 6, 8, 29
- Hong, Y., Zhang, K., Gu, J., Bi, S., Zhou, Y., Liu, D., Liu, F., Sunkavalli, K., Bui, T., and Tan, H. (2023). Lrm: Large reconstruction model for single image to 3d. *arXiv preprint arXiv:2311.04400*. 52
- Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., and Chen, W. (2021). Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*. 10, 12, 34
- Huang, S., Qi, S., Xiao, Y., Zhu, Y., Wu, Y. N., and Zhu, S.-C. (2018a). Cooperative holistic scene understanding: Unifying 3d object, layout, and camera pose estimation. In *Advances in Neural Information Processing Systems (NeurIPS)*. 69
- Huang, S., Qi, S., Zhu, Y., Xiao, Y., Xu, Y., and Zhu, S.-C. (2018b). Holistic 3d scene parsing and reconstruction from a single rgb image. In *Eur. Conf. Comput. Vis.* 69
- Huang, Z., Stojanov, S., Thai, A., Jampani, V., and Rehg, J. M. (2022). Planes vs. chairs: Category-guided 3d shape learning without any 3d cues. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part I*, pages 727–744. Springer. 5, 9
- Jain, A., Mildenhall, B., Barron, J. T., Abbeel, P., and Poole, B. (2022). Zero-shot text-guided object generation with dream fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 867–876. 6, 8, 9, 30, 52
- Jain, A., Tancik, M., and Abbeel, P. (2021). Putting nerf on a diet: Semantically consistent few-shot view synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5885–5894. 50
- Jang, W. and Agapito, L. (2021). Codenerf: Disentangled neural radiance fields for object categories. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12949–12958. 9
- Jetchev, N. (2021). Clipmatrix: Text-controlled creation of 3d textured meshes. *arXiv preprint arXiv:2109.12922*. 8
- Jiang, H., Jiang, Z., Grauman, K., and Zhu, Y. (2022). Few-view object reconstruction with unknown categories and camera poses. *arXiv preprint arXiv:2212.04492*. 48, 51, 61, 62, 63
- Jiang, H., Jiang, Z., Zhao, Y., and Huang, Q. (2023). Leap: Liberate sparse-view 3d modeling from camera poses. *arXiv preprint arXiv:2310.01410*. 48, 52
- Johari, M. M., Lepoittevin, Y., and Fleuret, F. (2022). Geonerf: Generalizing nerf with geometry priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18365–18375. 10, 15, 22, 30, 50

- Jun, H. and Nichol, A. (2023). Shap-e: Generating conditional 3d implicit functions. *arXiv preprint arXiv:2305.02463*. 10, 18, 19, 21, 23, 24, 29, 38, 40, 41, 61, 63
- Kabra, R., Burgess, C., Matthey, L., Kaufman, R. L., Greff, K., Reynolds, M., and Lerchner, A. (2019). Multi-object datasets. <https://github.com/deepmind/multi-object-datasets/>. 73
- Kanazawa, A., Tulsiani, S., Efros, A. A., and Malik, J. (2018). Learning category-specific mesh reconstruction from image collections. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 371–386. 5, 9, 29
- Karnewar, A., Vedaldi, A., Novotny, D., and Mitra, N. J. (2023). Holodiffusion: Training a 3d diffusion model using 2d images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18423–18433. 31, 50
- Khalid, N., Xie, T., Belilovsky, E., and Popa, T. (2022). Text to mesh without 3d supervision using limit subdivision. *arXiv preprint arXiv:2203.13333*. 8
- Kim, M., Seo, S., and Han, B. (2022). Infonerf: Ray entropy minimization for few-shot neural volume rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12912–12921. 48, 50
- Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*. 83
- Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A. C., Lo, W.-Y., et al. (2023). Segment anything. *arXiv preprint arXiv:2304.02643*. 17
- Kjellström, H., Romero, J., and Kragić, D. (2011). Visual object-action recognition: Inferring object affordances from human demonstration. *Computer Vision and Image Understanding*, 115(1):81–90. 72
- Kong, X., Liu, S., Lyu, X., Taher, M., Qi, X., and Davison, A. J. (2024). Eschernet: A generative model for scalable view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9503–9513. 50, 61, 62, 63
- Kuhn, H. W. (1955). The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97. 77
- Kulhánek, J., Derner, E., Sattler, T., and Babuška, R. (2022). Viewformer: Nerf-free neural rendering from few images using transformers. In *European Conference on Computer Vision*, pages 198–216. Springer. 10, 30, 50
- Lai, Z., Liu, S., Efros, A. A., and Wang, X. (2021). Video autoencoder: self-supervised disentanglement of static 3d structure and motion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9730–9740. 51

- Laine, S., Hellsten, J., Karras, T., Seol, Y., Lehtinen, J., and Aila, T. (2020). Modular primitives for high-performance differentiable rendering. *ACM Transactions on Graphics*, 39(6). 49, 59
- Land, M., Mennie, N., and Rusted, J. (1999). The roles of vision and eye movements in the control of activities of daily living. *Perception*, 28(11):1311–1328. 69
- Lee, H.-H. and Chang, A. X. (2022). Understanding pure clip guidance for voxel grid nerf models. *arXiv preprint arXiv:2209.15172*. 8, 30, 52
- Li, J., Tan, H., Zhang, K., Xu, Z., Luan, F., Xu, Y., Hong, Y., Sunkavalli, K., Shakhnarovich, G., and Bi, S. (2023a). Instant3d: Fast text-to-3d with sparse-view generation and large reconstruction model. *arXiv preprint arXiv:2311.06214*. 47, 48, 53, 54, 58
- Li, M., Duan, Y., Zhou, J., and Lu, J. (2023b). Diffusion-sdf: Text-to-shape via voxelized diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12642–12651. 29
- Liang, W., Zhao, Y., Zhu, Y., and Zhu, S.-C. (2016). What is where: Inferring containment relations from videos. In *IJCAI*. 72
- Lin, A., Zhang, J. Y., Ramanan, D., and Tulsiani, S. (2023a). Relpose++: Recovering 6d poses from sparse-view observations. *arXiv preprint arXiv:2305.04926*. 48, 52, 61, 62, 63
- Lin, C.-H., Gao, J., Tang, L., Takikawa, T., Zeng, X., Huang, X., Kreis, K., Fidler, S., Liu, M.-Y., and Lin, T.-Y. (2022). Magic3d: High-resolution text-to-3d content creation. *arXiv preprint arXiv:2211.10440*. 6, 8, 9
- Lin, C.-H., Gao, J., Tang, L., Takikawa, T., Zeng, X., Huang, X., Kreis, K., Fidler, S., Liu, M.-Y., and Lin, T.-Y. (2023b). Magic3d: High-resolution text-to-3d content creation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 300–309. 29, 47, 52
- Lin, C.-H., Ma, W.-C., Torralba, A., and Lucey, S. (2021). Barf: Bundle-adjusting neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5741–5751. 51
- Liu, M., Sheng, L., Yang, S., Shao, J., and Hu, S.-M. (2020). Morphing and sampling network for dense point cloud completion. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 11596–11603. 5
- Liu, M., Shi, R., Chen, L., Zhang, Z., Xu, C., Wei, X., Chen, H., Zeng, C., Gu, J., and Su, H. (2024a). One-2-3-45++: Fast single image to 3d objects with consistent multi-view generation and 3d diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10072–10083. 47, 49, 53, 58, 59

- Liu, M., Shi, R., Kuang, K., Zhu, Y., Li, X., Han, S., Cai, H., Porikli, F., and Su, H. (2023a). Openshape: Scaling up 3d shape representation towards open-world understanding. *Advances in neural information processing systems*, 36:44860–44879. 8, 60
- Liu, M., Sung, M., Mech, R., and Su, H. (2021). Deepmetahandles: Learning deformation meta-handles of 3d meshes with biharmonic coordinates. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12–21. 5, 29
- Liu, M., Xu, C., Jin, H., Chen, L., Varma T, M., Xu, Z., and Su, H. (2023b). One-2-3-45: Any single image to 3d mesh in 45 seconds without per-shape optimization. *Advances in Neural Information Processing Systems*, 36. 30, 38, 39, 42, 53, 58, 61, 62, 63
- Liu, M., Zeng, C., Wei, X., Shi, R., Chen, L., Xu, C., Zhang, M., Wang, Z., Zhang, X., Liu, I., et al. (2024b). Meshformer: High-quality mesh generation with 3d-guided reconstruction model. *Advances in Neural Information Processing Systems*. 46
- Liu, M., Zhu, Y., Cai, H., Han, S., Ling, Z., Porikli, F., and Su, H. (2023c). Partslip: Low-shot part segmentation for 3d point clouds via pretrained image-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21736–21746. 8
- Liu, R., Wu, R., Van Hoorick, B., Tokmakov, P., Zakharov, S., and Vondrick, C. (2023d). Zero-1-to-3: Zero-shot one image to 3d object. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9298–9309. 7, 10, 11, 12, 13, 17, 18, 19, 30, 38, 43, 44, 48, 52, 53, 54, 58
- Liu, Y., Lin, C., Zeng, Z., Long, X., Liu, L., Komura, T., and Wang, W. (2023e). Syncdreamer: Generating multiview-consistent images from a single-view image. *arXiv preprint arXiv:2309.03453*. 30, 31, 38, 43, 44, 48, 51, 53, 54, 58, 61, 63
- Liu, Y., Peng, S., Liu, L., Wang, Q., Wang, P., Theobalt, C., Zhou, X., and Wang, W. (2022a). Neural rays for occlusion-aware image-based rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7824–7833. 10, 30, 50
- Liu, Z., Dai, P., Li, R., Qi, X., and Fu, C.-W. (2022b). Iss: Image as stetting stone for text-guided 3d shape generation. *arXiv preprint arXiv:2209.04145*. 9
- Liu, Z., Dai, P., Li, R., Qi, X., and Fu, C.-W. (2023f). Iss++: Image as stepping stone for text-guided 3d shape generation. *arXiv preprint arXiv:2303.15181*. 9
- Liu, Z., Feng, Y., Black, M. J., Nowrouzezahrai, D., Paull, L., and Liu, W. (2023g). Meshdiffusion: Score-based generative 3d mesh modeling. *arXiv preprint arXiv:2303.08133*. 29

- Locatello, F., Weissenborn, D., Unterthiner, T., Mahendran, A., Heigold, G., Uszkoreit, J., Dosovitskiy, A., and Kipf, T. (2020). Object-centric learning with slot attention. *Advances in Neural Information Processing Systems (NeurIPS)*. 70, 73, 75, 77, 81, 86, 87
- Long, X., Guo, Y.-C., Lin, C., Liu, Y., Dou, Z., Liu, L., Ma, Y., Zhang, S.-H., Habermann, M., Theobalt, C., et al. (2024). Wonder3d: Single image to 3d using cross-domain diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9970–9980. 30, 31, 43, 44, 48, 51, 53, 54, 56, 57
- Long, X., Lin, C., Wang, P., Komura, T., and Wang, W. (2022). Sparseneus: Fast generalizable neural surface reconstruction from sparse views. In *European Conference on Computer Vision*, pages 210–227. Springer. 7, 10, 11, 14, 15, 22, 30, 42, 48, 50
- Lorensen, W. E. and Cline, H. E. (1987). Marching cubes: A high resolution 3d surface construction algorithm. *ACM siggraph computer graphics*, 21(4):163–169. 16
- Marr, D. (1982). Vision: A computational investigation into the human representation and processing of visual information. 1
- Mees, O., Tatarchenko, M., Brox, T., and Burgard, W. (2019). Self-supervised 3d shape and viewpoint estimation from single images for robotics. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6083–6089. IEEE. 5
- Melas-Kyriazi, L., Laina, I., Rupprecht, C., and Vedaldi, A. (2023a). Realfusion: 360deg reconstruction of any object from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8446–8455. 7, 9, 10, 18, 19, 20, 21, 30, 52
- Melas-Kyriazi, L., Rupprecht, C., and Vedaldi, A. (2023b). pc2: Projection—conditioned point cloud diffusion for single-image 3d reconstruction. *arXiv preprint arXiv:2302.10668*. 9
- Melas-Kyriazi, L., Rupprecht, C., and Vedaldi, A. (2023c). Pc2: Projection-conditioned point cloud diffusion for single-image 3d reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12923–12932. 29
- Mescheder, L., Oechsle, M., Niemeyer, M., Nowozin, S., and Geiger, A. (2019). Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4460–4470. 9, 29, 75
- Metzer, G., Richardson, E., Patashnik, O., Giryes, R., and Cohen-Or, D. (2023). Latent-nerf for shape-guided generation of 3d shapes and textures. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12663–12673. 9, 10, 30, 52

- Michel, O., Bar-On, R., Liu, R., Benaim, S., and Hanocka, R. (2022). Text2mesh: Text-driven neural stylization for meshes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13492–13502. 6, 8, 9, 30, 52
- Mildenhall, B., Srinivasan, P. P., Tancik, M., Barron, J. T., Ramamoorthi, R., and Ng, R. (2020). Nerf: Representing scenes as neural radiance fields for view synthesis. In *Eur. Conf. Comput. Vis.* 73
- Mildenhall, B., Srinivasan, P. P., Tancik, M., Barron, J. T., Ramamoorthi, R., and Ng, R. (2021). Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106. 7, 10, 13, 47
- Mittal, P., Cheng, Y.-C., Singh, M., and Tulsiani, S. (2022). Autosdf: Shape priors for 3d completion, reconstruction and generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 306–315. 9
- Mo, K., Guerrero, P., Yi, L., Su, H., Wonka, P., Mitra, N., and Guibas, L. J. (2019a). Structurenets: Hierarchical graph networks for 3d shape generation. *arXiv preprint arXiv:1908.00575*. 73
- Mo, K., Guibas, L. J., Mukadam, M., Gupta, A., and Tulsiani, S. (2021). Where2act: From pixels to actions for articulated 3d objects. In *Int. Conf. Comput. Vis.* 72
- Mo, K., Qin, Y., Xiang, F., Su, H., and Guibas, L. (2022). O2o-afford: Annotation-free large-scale object-object affordance learning. In *Conference on Robot Learning*, pages 1666–1677. PMLR. 72
- Mo, K., Zhu, S., Chang, A. X., Yi, L., Tripathi, S., Guibas, L. J., and Su, H. (2019b). Partnet: A large-scale benchmark for fine-grained and hierarchical part-level 3d object understanding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 909–918. 71, 72, 78, 79
- Mohammad Khalid, N., Xie, T., Belilovsky, E., and Popa, T. (2022). Clip-mesh: Generating textured meshes from text using pretrained image-text models. In *SIGGRAPH Asia 2022 conference papers*, pages 1–8. 30, 52
- Müller, N., Simonelli, A., Porzi, L., Bulò, S. R., Nießner, M., and Kotschieder, P. (2022). Autorf: Learning 3d object radiance fields from single view observations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3971–3980. 9
- Myers, A., Teo, C. L., Fermüller, C., and Aloimonos, Y. (2015). Affordance detection of tool parts from geometric features. In *International Conference on Robotics and Automation (ICRA)*, pages 1374–1381. IEEE. 72
- Nagarajan, T., Feichtenhofer, C., and Grauman, K. (2019). Grounded human-object interaction hotspots from video. In *Int. Conf. Comput. Vis.* 72

- Nagarajan, T. and Grauman, K. (2020). Learning affordance landscapes for interaction exploration in 3d environments. *Advances in Neural Information Processing Systems (NeurIPS)*. 72
- Nagarajan, T., Li, Y., Feichtenhofer, C., and Grauman, K. (2020). Ego-topo: Environment affordances from egocentric video. In *IEEE Conf. Comput. Vis. Pattern Recog.* 72
- Nash, C., Ganin, Y., Eslami, S. A., and Battaglia, P. (2020). Polygen: An autoregressive generative model of 3d meshes. In *International conference on machine learning*, pages 7220–7229. PMLR. 9, 29
- Nguyen, A., Kanoulas, D., Caldwell, D. G., and Tsagarakis, N. G. (2017). Object-based affordances detection with convolutional neural networks and dense conditional random fields. In *International Conference on Intelligent Robots and Systems (IROS)*. 72
- Nichol, A., Jun, H., Dhariwal, P., Mishkin, P., and Chen, M. (2022). Point-e: A system for generating 3d point clouds from complex prompts. *arXiv preprint arXiv:2212.08751*. 10, 18, 19, 21, 29
- Niemeyer, M., Barron, J. T., Mildenhall, B., Sajjadi, M. S., Geiger, A., and Radwan, N. (2022). Regnerf: Regularizing neural radiance fields for view synthesis from sparse inputs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5480–5490. 50
- Park, J. J., Florence, P., Straub, J., Newcombe, R., and Lovegrove, S. (2019). Deepsdf: Learning continuous signed distance functions for shape representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 165–174. 9, 29
- Paschalidou, D., Gool, L. V., and Geiger, A. (2020). Learning unsupervised hierarchical part decomposition of 3d objects from a single rgb image. In *IEEE Conf. Comput. Vis. Pattern Recog.* 73
- Paschalidou, D., Katharopoulos, A., Geiger, A., and Fidler, S. (2021). Neural parts: Learning expressive 3d shape abstractions with invertible neural networks. In *IEEE Conf. Comput. Vis. Pattern Recog.* 73
- Paschalidou, D., Ulusoy, A. O., and Geiger, A. (2019). Superquadrics revisited: Learning 3d shape parsing beyond cuboids. In *IEEE Conf. Comput. Vis. Pattern Recog.* 73
- Pavlakos, G., Choutas, V., Ghorbani, N., Bolkart, T., Osman, A. A., Tzionas, D., and Black, M. J. (2019). Expressive body capture: 3d hands, face, and body from a single image. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10975–10985. 9
- Poole, B., Jain, A., Barron, J. T., and Mildenhall, B. (2022). Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*. 6, 7, 8, 9, 20, 23, 29, 38, 41, 47, 52

- Qian, G., Mai, J., Hamdi, A., Ren, J., Siarohin, A., Li, B., Lee, H.-Y., Skorokhodov, I., Wonka, P., Tulyakov, S., et al. (2023). Magic123: One image to high-quality 3d object generation using both 2d and 3d diffusion priors. *arXiv preprint arXiv:2306.17843*. 30, 53
- Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. (2021). Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR. 5, 8, 29, 52
- Raj, A., Kaza, S., Poole, B., Niemeyer, M., Ruiz, N., Mildenhall, B., Zada, S., Aberman, K., Rubinstein, M., Barron, J., et al. (2023). Dreambooth3d: Subject-driven text-to-3d generation. *arXiv preprint arXiv:2303.13508*. 9, 30, 53
- Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., and Chen, M. (2022). Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*. 5, 8, 23
- Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., and Sutskever, I. (2021). Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pages 8821–8831. PMLR. 5, 8, 12, 29, 47, 52
- Reizenstein, J., Shapovalov, R., Henzler, P., Sbordone, L., Labatut, P., and Novotny, D. (2021). Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction. In *International Conference on Computer Vision*. 9, 10
- Rematas, K., Martin-Brualla, R., and Ferrari, V. (2021). Sharf: Shape-conditioned radiance fields from a single view. *arXiv preprint arXiv:2102.08860*. 50
- Ren, Y., Wang, F., Zhang, T., Pollefeys, M., and Süssstrunk, S. (2022). Volrecon: Volume rendering of signed ray distance functions for generalizable multi-view reconstruction. *arXiv preprint arXiv:2212.08067*. 11
- Ren, Y., Zhang, T., Pollefeys, M., Süssstrunk, S., and Wang, F. (2023). Volrecon: Volume rendering of signed ray distance functions for generalizable multi-view reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16685–16695. 30, 50
- Ren, Z., Misra, I., Schwing, A. G., and Girdhar, R. (2021). 3d spatial recognition without spatially labeled 3d. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 13204–13213. 73
- Richardson, E., Metzger, G., Alaluf, Y., Giryes, R., and Cohen-Or, D. (2023). Texture: Text-guided texturing of 3d shapes. *arXiv preprint arXiv:2302.01721*. 9, 30
- Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. (2022). High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695. 5, 7, 8, 12, 18, 23, 29, 47, 48, 52, 53, 54

- Roy, A. and Todorovic, S. (2016). A multi-scale cnn for affordance segmentation in rgb images. In *Eur. Conf. Comput. Vis.* 72
- Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E. L., Ghasemipour, K., Gontijo Lopes, R., Karagol Ayan, B., Salimans, T., et al. (2022). Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35:36479–36494. 5, 8, 10, 12, 29, 47, 52
- Saito, S., Huang, Z., Natsume, R., Morishima, S., Kanazawa, A., and Li, H. (2019). Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2304–2314. 9
- Sajjadi, M. S., Meyer, H., Pot, E., Bergmann, U., Greff, K., Radwan, N., Vora, S., Lučić, M., Duckworth, D., Dosovitskiy, A., et al. (2022). Scene representation transformer: Geometry-free novel view synthesis through set-latent scene representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6229–6238. 52
- Sanghi, A., Chu, H., Lambourne, J. G., Wang, Y., Cheng, C.-Y., Fumero, M., and Malekshan, K. R. (2022). Clip-forge: Towards zero-shot text-to-shape generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18603–18613. 9, 29
- Schonberger, J. L. and Frahm, J.-M. (2016). Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4104–4113. 48, 56
- Schönberger, J. L., Zheng, E., Frahm, J.-M., and Pollefeys, M. (2016). Pixelwise view selection for unstructured multi-view stereo. In *Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part III 14*, pages 501–518. Springer. 47
- Seo, J., Jang, W., Kwak, M.-S., Ko, J., Kim, H., Kim, J., Kim, J.-H., Lee, J., and Kim, S. (2023). Let 2d diffusion model know 3d-consistency for robust text-to-3d generation. *arXiv preprint arXiv:2303.07937*. 7, 9, 10, 18, 19, 23, 30, 53
- Shepard, R. N. and Metzler, J. (1971). Mental rotation of three-dimensional objects. *Science*, 171(3972):701–703. 1
- Shi, R., Chen, H., Zhang, Z., Liu, M., Xu, C., Wei, X., Chen, L., Zeng, C., and Su, H. (2023a). Zero123++: a single image to consistent multi-view diffusion base model. *arXiv preprint arXiv:2310.15110*. 10, 48, 53, 54, 55
- Shi, R., Wei, X., Wang, C., and Su, H. (2023b). Zerorf: Fast sparse view 360 $\{\backslash\deg\}$ reconstruction with zero pretraining. *arXiv preprint arXiv:2312.09249*. 50

- Shi, Y., Wang, P., Ye, J., Long, M., Li, K., and Yang, X. (2023c). Mvdream: Multi-view diffusion for 3d generation. *arXiv preprint arXiv:2308.16512*. 30, 31, 40, 41, 47, 48, 51, 53, 54
- Sinha, S., Zhang, J. Y., Tagliasacchi, A., Gilitschenski, I., and Lindell, D. B. (2023). Sparsepose: Sparse-view camera pose regression and refinement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21349–21359. 48, 51
- Soatto, S. (2013). Actionable information in vision. In *Machine Learning for Computer Vision*, pages 17–48. Springer. 69
- Stelzner, K., Kersting, K., and Kosiorek, A. R. (2021). Decomposing 3d scenes into objects via unsupervised volume segmentation. *arXiv preprint arXiv:2104.01148*. 73
- Stereopsis, R. M. (2010). Accurate, dense, and robust multiview stereopsis. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, 32(8). 47
- Sun, C.-Y., Zou, Q.-F., Tong, X., and Liu, Y. (2019). Learning adaptive hierarchical cuboid abstractions of 3d shape collections. *ACM Trans. Graph.*, 38(6):1–13. 73
- Tang, H., Yang, S., Liu, Z., Hong, K., Yu, Z., Li, X., Dai, G., Wang, Y., and Han, S. (2023a). Torchsparse++: Efficient training and inference framework for sparse convolution on gpus. In *IEEE/ACM International Symposium on Microarchitecture (MICRO)*. 35
- Tang, J., Chen, Z., Chen, X., Wang, T., Zeng, G., and Liu, Z. (2024). Lgm: Large multi-view gaussian model for high-resolution 3d content creation. *arXiv preprint arXiv:2402.05054*. 58
- Tang, J., Ren, J., Zhou, H., Liu, Z., and Zeng, G. (2023b). Dreamgaussian: Generative gaussian splatting for efficient 3d content creation. *arXiv preprint arXiv:2309.16653*. 30, 38, 53, 58, 61, 63
- Tang, J., Wang, T., Zhang, B., Zhang, T., Yi, R., Ma, L., and Chen, D. (2023c). Make-it-3d: High-fidelity 3d creation from a single image with diffusion prior. *arXiv preprint arXiv:2303.14184*. 9, 30, 53
- Tewari, A., Yin, T., Cazenavette, G., Rezhikov, S., Tenenbaum, J. B., Durand, F., Freeman, W. T., and Sitzmann, V. (2023). Diffusion with forward models: Solving stochastic inverse problems without direct supervision. *arXiv preprint arXiv:2306.11719*. 31, 50
- Tochilkin, D., Pankratz, D., Liu, Z., Huang, Z., Letts, A., Li, Y., Liang, D., Laforte, C., Jampani, V., and Cao, Y.-P. (2024). Tripopr: Fast 3d object reconstruction from a single image. 62

- Toussaint, M., Allen, K. R., Smith, K. A., and Tenenbaum, J. B. (2019). Differentiable physics and stable modes for tool-use and manipulation planning-extended abstract. In *IJCAI*, pages 6231–6235. 88
- Trevithick, A. and Yang, B. (2021). Grf: Learning a general radiance field for 3d representation and rendering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15182–15192. 10, 30, 50
- Truong, P., Rakotosaona, M.-J., Manhardt, F., and Tombari, F. (2023). Sparf: Neural radiance fields from sparse and noisy poses. In *CVF Conference on Computer Vision and Pattern Recognition, CVPR*, volume 1. 51
- Tulsiani, S., Su, H., Guibas, L. J., Efros, A. A., and Malik, J. (2017). Learning shape abstractions by assembling volumetric primitives. In *IEEE Conf. Comput. Vis. Pattern Recog.* 73
- Tung, H.-Y. F., Cheng, R., and Fragkiadaki, K. (2019). Learning spatial common sense with geometry-aware recurrent networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2595–2603. 51
- Van Steenkiste, S., Chang, M., Greff, K., and Schmidhuber, J. (2018). Relational neural expectation maximization: Unsupervised discovery of objects and their interactions. *arXiv preprint arXiv:1802.10353*. 73
- Varma, M., Wang, P., Chen, X., Chen, T., Venugopalan, S., and Wang, Z. (2022). Is attention all that nerf needs? In *The Eleventh International Conference on Learning Representations*. 10
- Wang, H., Du, X., Li, J., Yeh, R. A., and Shakhnarovich, G. (2022a). Score jacobian chaining: Lifting pretrained 2d diffusion models for 3d generation. *arXiv preprint arXiv:2212.00774*. 9
- Wang, H., Du, X., Li, J., Yeh, R. A., and Shakhnarovich, G. (2023a). Score jacobian chaining: Lifting pretrained 2d diffusion models for 3d generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12619–12629. 30, 53
- Wang, H., Liang, W., and Yu, L.-F. (2017a). Transferring objects: Joint inference of container and human pose. In *Int. Conf. Comput. Vis.* 72
- Wang, H., Sridhar, S., Huang, J., Valentin, J., Song, S., and Guibas, L. J. (2019). Normalized object coordinate space for category-level 6d object pose and size estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2642–2651. 48, 56
- Wang, J., Rupprecht, C., and Novotny, D. (2023b). Posediffusion: Solving pose estimation via diffusion-aided bundle adjustment. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9773–9783. 52

- Wang, N., Zhang, Y., Li, Z., Fu, Y., Liu, W., and Jiang, Y.-G. (2018). Pixel2mesh: Generating 3d mesh models from single rgb images. In *Proceedings of the European conference on computer vision (ECCV)*, pages 52–67. 9, 29
- Wang, P., Chen, X., Chen, T., Venugopalan, S., Wang, Z., et al. (2022b). Is attention all nerf needs? *arXiv preprint arXiv:2207.13298*. 30, 50
- Wang, P., Liu, L., Liu, Y., Theobalt, C., Komura, T., and Wang, W. (2021a). Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *arXiv preprint arXiv:2106.10689*. 10, 13, 15, 16, 47
- Wang, P., Tan, H., Bi, S., Xu, Y., Luan, F., Sunkavalli, K., Wang, W., Xu, Z., and Zhang, K. (2023c). Pf-lrm: Pose-free large reconstruction model for joint pose and shape prediction. *arXiv preprint arXiv:2311.12024*. 52
- Wang, Q., Wang, Z., Genova, K., Srinivasan, P. P., Zhou, H., Barron, J. T., Martin-Brualla, R., Snavely, N., and Funkhouser, T. (2021b). Ibrnet: Learning multi-view image-based rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4690–4699. 10, 30, 50
- Wang, X., Girdhar, R., and Gupta, A. (2017b). Binge watching: Scaling affordance learning from sitcoms. In *IEEE Conf. Comput. Vis. Pattern Recog.* 72
- Wang, Y., Wu, R., Mo, K., Ke, J., Fan, Q., Guibas, L. J., and Dong, H. (2022c). Adaafford: Learning to adapt manipulation affordance for 3d articulated objects via few-shot interactions. In *Eur. Conf. Comput. Vis.*, pages 90–107. Springer. 72
- Wang, Z., Lu, C., Wang, Y., Bao, F., Li, C., Su, H., and Zhu, J. (2023d). Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. *arXiv preprint arXiv:2305.16213*. 30, 40, 41, 47, 52
- Wang, Z., Wu, S., Xie, W., Chen, M., and Prisacariu, V. A. (2021c). Nerf-: Neural radiance fields without known camera parameters. *arXiv preprint arXiv:2102.07064*. 51
- Wei, J., Lin, G., Yap, K.-H., Hung, T.-Y., and Xie, L. (2020). Multi-path region mining for weakly supervised 3d semantic segmentation on point clouds. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 4384–4393. 73
- Wei, J., Wang, H., Feng, J., Lin, G., and Yap, K.-H. (2023). Taps3d: Text-guided 3d textured shape generation from pseudo supervision. 9
- Wen, C., Zhang, Y., Li, Z., and Fu, Y. (2019). Pixel2mesh++: Multi-view 3d mesh generation via deformation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1042–1051. 5, 9
- Weng, H., Yang, T., Wang, J., Li, Y., Zhang, T., Chen, C., and Zhang, L. (2023). Consistent123: Improve consistency for one image to 3d object synthesis. *arXiv preprint arXiv:2310.08092*. 30, 53

- Wu, C.-H., Chen, Y.-C., Solarte, B., Yuan, L., and Sun, M. (2023a). ifusion: Inverting diffusion for pose-free reconstruction from sparse views. [50](#), [52](#), [61](#), [62](#), [63](#)
- Wu, C.-Y., Johnson, J., Malik, J., Feichtenhofer, C., and Gkioxari, G. (2023b). Multiview compressive coding for 3d reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9065–9075. [9](#), [29](#)
- Wu, J., Wang, Y., Xue, T., Sun, X., Freeman, B., and Tenenbaum, J. (2017). Marrnet: 3d shape reconstruction via 2.5 d sketches. *Advances in neural information processing systems*, 30. [9](#), [29](#)
- Wu, R., Mildenhall, B., Henzler, P., Park, K., Gao, R., Watson, D., Srinivasan, P. P., Verbin, D., Barron, J. T., Poole, B., et al. (2024). Reconfusion: 3d reconstruction with diffusion priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21551–21561. [50](#)
- Wu, T., Zhang, J., Fu, X., Wang, Y., Ren, J., Pan, L., Wu, W., Yang, L., Wang, J., Qian, C., et al. (2023c). Omniobject3d: Large-vocabulary 3d object dataset for realistic perception, reconstruction and generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 803–814. [61](#), [62](#), [63](#), [66](#)
- Xia, Y., Tang, H., Timofte, R., and Van Gool, L. (2022). Sinerf: Sinusoidal neural radiance fields for joint pose estimation and scene reconstruction. *arXiv preprint arXiv:2210.04553*. [51](#)
- Xiang, F., Qin, Y., Mo, K., Xia, Y., Zhu, H., Liu, F., Liu, M., Jiang, H., Yuan, Y., Wang, H., et al. (2020). Sapien: A simulated part-based interactive environment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11097–11107. [71](#), [78](#), [79](#)
- Xie, D., Todorovic, S., and Zhu, S.-C. (2013). Inferring” dark matter” and” dark energy” from videos. In *Int. Conf. Comput. Vis.* [72](#)
- Xie, H., Yao, H., Sun, X., Zhou, S., and Zhang, S. (2019). Pix2vox: Context-aware 3d reconstruction from single and multi-view images. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2690–2698. [9](#), [29](#)
- Xie, H., Yao, H., Zhang, S., Zhou, S., and Sun, W. (2020). Pix2vox++: Multi-scale context-aware 3d object reconstruction from single and multiple images. *International Journal of Computer Vision*, 128(12):2919–2935. [5](#), [9](#)
- Xu, C., Chen, Y., Wang, H., Zhu, S.-C., Zhu, Y., and Huang, S. (2022). Partafford: Part-level affordance discovery from 3d objects. *arXiv preprint arXiv:2202.13519*. [89](#)
- Xu, C., Li, A., Chen, L., Liu, Y., Shi, R., Su, H., and Liu, M. (2024). Sparp: Fast 3d object reconstruction and pose estimation from sparse views. In *European Conference on Computer Vision*, pages 143–163. Springer. [68](#)

- Xu, D., Jiang, Y., Wang, P., Fan, Z., Wang, Y., and Wang, Z. (2023a). Neurallift-360: Lifting an in-the-wild 2d photo to a 3d object with 360deg views. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4479–4489. [9](#), [30](#), [53](#)
- Xu, J., Wang, X., Cheng, W., Cao, Y.-P., Shan, Y., Qie, X., and Gao, S. (2023b). Dream3d: Zero-shot text-to-3d synthesis using 3d shape prior and text-to-image diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20908–20918. [9](#), [30](#), [53](#)
- Xu, Q., Wang, W., Ceylan, D., Mech, R., and Neumann, U. (2019). Disn: Deep implicit surface network for high-quality single-view 3d reconstruction. *Advances in neural information processing systems*, 32. [9](#), [29](#)
- Yagubbayli, F., Wang, Y., Tonioni, A., and Tombari, F. (2021). Legoformer: Transformers for block-by-block multi-view 3d reconstruction. *arXiv preprint arXiv:2106.12102*. [5](#), [9](#)
- Yang, D., Tosun, T., Eisner, B., Isler, V., and Lee, D. (2021). Robotic grasping through combined image-based grasp proposal and 3d reconstruction. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6350–6356. IEEE. [5](#)
- Yang, H., Hong, L., Li, A., Hu, T., Li, Z., Lee, G. H., and Wang, L. (2023). Contranerf: Generalizable neural radiance fields for synthetic-to-real novel view synthesis via contrastive learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16508–16517. [10](#), [30](#), [50](#)
- Yang, J., Mo, K., Lai, Y.-K., Guibas, L. J., and Gao, L. (2020). Dsg-net: Disentangled structured mesh net for controllable generation of fine geometry. *arXiv preprint arXiv:2008.05440*. [73](#)
- Yang, K. and Chen, X. (2021). Unsupervised learning for cuboid shape abstraction via joint segmentation from point clouds. *ACM Trans. Graph.* [73](#), [77](#)
- Yang, Y., Feng, C., Shen, Y., and Tian, D. (2018). Foldingnet: Point cloud auto-encoder via deep grid deformation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 206–215. [9](#)
- Yang, Z., Ren, Z., Bautista, M. A., Zhang, Z., Shan, Q., and Huang, Q. (2022). Fvor: Robust joint shape and pose optimization for few-view object reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2497–2507. [51](#)
- Yariv, L., Gu, J., Kasten, Y., and Lipman, Y. (2021). Volume rendering of neural implicit surfaces. *Advances in Neural Information Processing Systems*, 34:4805–4815. [47](#)

- Ye, J., Wang, P., Li, K., Shi, Y., and Wang, H. (2023). Consistent-1-to-3: Consistent image to 3d view synthesis via geometry-aware diffusion models. *arXiv preprint arXiv:2310.03020*. 30, 53
- Yu, A., Ye, V., Tancik, M., and Kanazawa, A. (2021a). pixelnerf: Neural radiance fields from one or few images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4578–4587. 10, 50
- Yu, C., Zhou, Q., Li, J., Zhang, Z., Wang, Z., and Wang, F. (2023a). Points-to-3d: Bridging the gap between sparse points and shape-controllable text-to-3d generation. *arXiv preprint arXiv:2307.13908*. 30, 53
- Yu, H.-X., Guibas, L. J., and Wu, J. (2021b). Unsupervised discovery of object radiance fields. *arXiv preprint arXiv:2107.07905*. 73
- Yu, L.-F., Duncan, N., and Yeung, S.-K. (2015). Fill and transfer: A simple physics-based approach for containability reasoning. In *Int. Conf. Comput. Vis.* 72
- Yu, W., Qian, X., Huo, J., Huang, T., Zhao, B., and Fu, Y. (2023b). Pushing the limits of 3d shape generation at scale. *arXiv preprint arXiv:2306.11510*. 29
- Zeng, X., Vahdat, A., Williams, F., Gojcic, Z., Litany, O., Fidler, S., and Kreis, K. (2022). Lion: Latent point diffusion models for 3d shape generation. *arXiv preprint arXiv:2210.06978*. 9, 29
- Zhang, B., Tang, J., Niessner, M., and Wonka, P. (2023). 3dshape2vecset: A 3d shape representation for neural fields and generative diffusion models. *arXiv preprint arXiv:2301.11445*. 29
- Zhang, J. Y., Ramanan, D., and Tulsiani, S. (2022a). Relpose: Predicting probabilistic relative rotation for single objects in the wild. In *European Conference on Computer Vision*, pages 592–611. Springer. 48, 52
- Zhang, L. (2023). Reference-only control. <https://github.com/Mikubill/sd-webui-controlnet/discussions/1236>. 33, 55
- Zhang, L. and Agrawala, M. (2023). Adding conditional control to text-to-image diffusion models. 12
- Zhang, X., Bi, S., Sunkavalli, K., Su, H., and Xu, Z. (2022b). Nerfusion: Fusing radiance fields for large-scale scene reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5449–5458. 10
- Zhao, Y. and Zhu, S.-C. (2013). Scene parsing by integrating function, geometry and appearance models. In *IEEE Conf. Comput. Vis. Pattern Recog.* 72
- Zhao, Z., Liu, W., Chen, X., Zeng, X., Wang, R., Cheng, P., Fu, B., Chen, T., Yu, G., and Gao, S. (2023). Michelangelo: Conditional 3d shape generation based on shape-image-text aligned latent representation. *arXiv preprint arXiv:2306.17115*. 29

Zheng, X.-Y., Pan, H., Wang, P.-S., Tong, X., Liu, Y., and Shum, H.-Y. (2023). Locally attentional sdf diffusion for controllable 3d shape generation. *arXiv preprint arXiv:2305.04461*. 29, 35

Zhou, Z. and Tulsiani, S. (2023). Sparsefusion: Distilling view-conditioned diffusion for 3d reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12588–12597. 9, 31, 48, 50

Zhu, Y., Gao, T., Fan, L., Huang, S., Edmonds, M., Liu, H., Gao, F., Zhang, C., Qi, S., Wu, Y. N., et al. (2020). Dark, beyond deep: A paradigm shift to cognitive ai with humanlike common sense. *Engineering*, 6(3):310–345. 72

Zhu, Y., Jiang, C., Zhao, Y., Terzopoulos, D., and Zhu, S.-C. (2016). Inferring forces and learning human utilities from videos. In *IEEE Conf. Comput. Vis. Pattern Recog.* 72

Zhu, Y., Zhao, Y., and Zhu, S.-C. (2015). Understanding tools: Task-oriented object modeling, learning and recognition. In *IEEE Conf. Comput. Vis. Pattern Recog.* 72

Zuffi, S., Kanazawa, A., and Black, M. J. (2018). Lions and tigers and bears: Capturing non-rigid, 3d, articulated shape from images. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 3955–3963. 9

Zuffi, S., Kanazawa, A., Jacobs, D. W., and Black, M. J. (2017). 3d menagerie: Modeling the 3d shape and pose of animals. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6365–6373. 9