Learning 3D Information from Large Image Collections

Ta-Ying Cheng

St. Catherine's College

University of Oxford

Supervisor: Niki Trigoni, Andrew Markham

submitted for the degree of

D.Phil. in Computer Science

Michaelmas Term 2024

Acknowledgements

Reflecting on the past three years after I embarked on this D.Phil. journey is exciting, memorable, and humbling.

First and foremost, I am extremely grateful to my supervisors, Niki Trigoni and Andrew Markham, for their generous support towards my research in 3D-aware content creation. They offered me countless valuable advice, academically and beyond, that shaped me into the researcher I am today. Niki's unequivocal work ethic and creativity in problem-solving are the utmost inspiration to me, and I will always keep in mind Andrew's advice to "look 5 years into the future and think about what remains as impactful research".

I would like to thank my internship mentor, Matheus Gadelha, for my first two years at Adobe Research. His advice, from problem formulation and experiment design to paper writing, helped me in building the fundamentals to pursue 3D computer vision research. The experiences in these Adobe internships, with the guidance from all my co-mentors Thibault Groueix, Soren Pirk, Radomir Mech, and Matt Fisher, are unforgettable, to say the least. They also led me to a third internship at Adobe Research London, where I got to work with friendly, supportive, and insightful mentors Chun-Hao Huang and Duygu Ceylan for the final part of my study.

It was also extremely fortunate to have met Varun Jampani at the later stage of my D.Phil. through my collaborations with Stability AI. His vision of the important problems in computer vision and valuable advice on problem formulation constantly push me to be the best version of myself every day.

The D.Phil. journey would be incomplete without my inspiring collaborators: Chun-Hsiao Yeh, Prafull Sharma, Chuan-En Lin, Qian Xie, and Chenyang Ma.

Thank you for the constructive criticisms and constant encouragements.

The three dream-like years in the UK would lose many of its vibrant colours without my most supportive friends at Oxford: Albert Huang, Han Lee, Allen Lee, Julie Lin, Daqian Shao. I would never forget the constant laughter and joy shared across the most unorthodox trips, unexpected afternoon drinks, or spontaneous weekend dinners.

To all the friends I made from high school, college, and internships: Sharlene Chen, Ryan Sun, Frank Yang, Steven Liu, Karran Pandey, Julia Guerrero Viu, it is a blessing to meet you all.

I am indebted to my parents who wholeheartedly supported my decision to pursue this degree. I am also particularly thankful to the company of Jocelyn Chen, for the past 3 years in the UK. Moving to a city 16 hours away from where I call home is difficult, but all of you made it a little easier. I hope I have made you proud.

Abstract

Photos and videos, the most popular ways for us to capture the environment around us, are 2D-pixel representations that contain implicit yet rich 3D information.

As 2D images are much easier to capture than 3D data, the past decade of technological advance has catalyzed the creation of image datasets that are much larger and more diverse compared to their 3D counterparts. This has led to significant improvements in 2D image recognition and generation tasks but much more limited improvements in 3D-aware computer vision problems.

In this thesis, we attempt to isolate and extract 3D information from large image datasets with very little 3D data for assistance. Specifically, we explore large image-pretrained models, both for recognition and generation tasks, and focus on how we can extract three types of 3D information: 1) geometry 2) continuous movement-based attributes (e.g., camera motion, time-of-day lighting, non-rigid object motion), and 3) materials.

In Chapter 3, we present 3DMiner, an end-to-end pipeline to obtain geometry from a large set of unannotated image collections. In Chapter 4, we present Continuous 3D Words, a way to extract continuous, 3D-aware motions like time-of-day illumination or camera parameters and further control them during image generation and editing. In Chapter 5, we show that generative models trained on large image datasets can implicitly extract and transfer materials from one exemplar to another image, without the need for any further finetuning.

Overall, this thesis shows that, with minimal-to-none 3D data and model training, these 3D-aware attributes can be disentangled from the complex information presented in images. The resulting features are beneficial to a wide range of generation and reconstruction tasks.

Table of Contents

1 | Introduction

1.1 Motivation

Photos and videos are widely used and loved methods for capturing our 3D world. Whilst just being represented as a flat, two-dimensional image plane, they also imply 3D information which we, as humans, can extract by simply looking at it. For example, given an image of a living room, we can easily estimate the shape of the chair, the material of the wooden table, and the light source coming from the windows.

For the past decade, the increasing ubiquity of technology has enabled us to accumulate large-scale image data with exponentially rising volumes, and many deep learning networks targeting discriminative and generative tasks have greatly benefited from these readily available datasets. As these deep models mature, we realise that many of the learnt image features also implicitly encode these 3D information — DinoViT (Caron et al., 2021) has shown its capability to correspond parts between objects of the same species; Stable Diffusion (Rombach, Blattmann, Lorenz, Esser, & Ommer, 2021) can generate photorealistic images informed by physical properties like illumination.

A plethora of models have shown capabilities to obtain image features highly useful for performing fine-grained recognition tasks in the 2D-pixel space (Kirillov et al., 2023; Ravi et al., 2024). However, there remain various concepts, in particular attributes that can only be reasoned about in 3D, that are challenging to disentangle within the feature space of a given image. For example, it would be very difficult to disentangle object geometry from its background and textures, light reflections from material properties, or camera changes from estimated depths without having

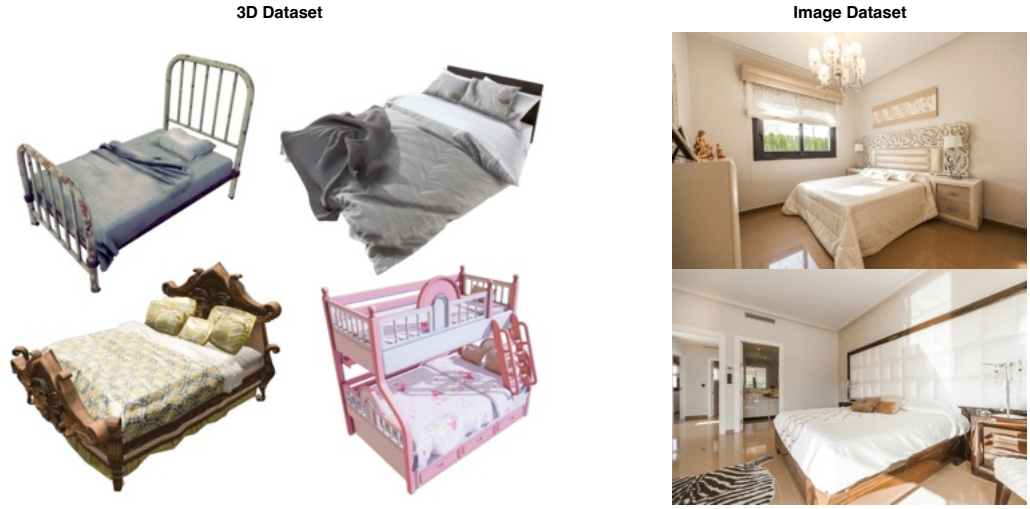

Figure 1.1: Diversity Difference between 2D and 3D Datasets. 3D datasets are often in similar styles, single object, and does not contain image backgrounds. 2D datasets, on the other hand, often provide much more complexity in terms of composition.

awareness of the entire 3D scene. To fully utilize the 3D reasoning capabilities of these 2D-pretrained models, we would like to better understand if we can model these attributes.

One straightforward intuition to distill these independent attributes is to model them in 3D with physical rendering engines and then learn from their renderings. For instance, one can learn from a large set of a bird's wing flapping from multiple views with camera parameters. This can then be used to extract the wing motion from object shapes which can then be used as part of the conditions for image generation of a different bird. Another example would be to render the same object from multiple views with different materials and aim to learn a representation that disentangles the materials from the geometry of the object.

The main challenge of this problem setup lies in the lack of diversity in 3D datasets. While the computer vision and deep learning explosion have significantly increased the efforts on 3D dataset collection, the progression in a variety of 3D

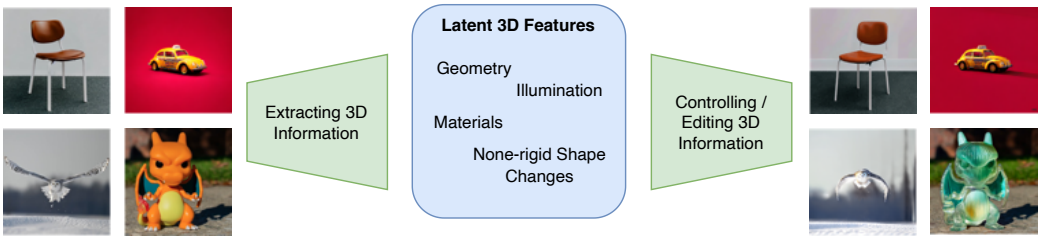

Figure 1.2: A general framework for learning 3D information from large image collections. In our problem formulation, we aim to extract latent features containing features of independent and disentangled 3D properties (e.g., geometry, illumination, materials, non-rigid shape deformations). These features can then be used for downstream tasks such as controlling generative models.

datasets is at present much more limited. The current largest 3D object dataset, Objaverse-XL (Deitke et al., 2023, 2024), comprises 10 million objects, which is in fact on par with a 2D image dataset in terms of scale (each object can be rendered into images from multiple viewpoints). However, while image datasets like LAION (Schuhmann et al., 2022) contain images of various compositions, backgrounds, and even painting/artistic styles, most Objaverse-XL data remain single-object oriented with similar styles and not much composition (see Figure 1.1 for examples), not to mention the even more limited data to model attributes like the aforementioned bird's flying motion. The time-consuming process of 3D data collection, either through capturing actual 3D objects or designing them via animation software, limits the available ground truths in supervising the task of disentangling 3D-aware concepts.

Another challenge related to the extraction of these 3D-aware information is how to evaluate the extracted features and what applications they have. For explicit attributes like geometry, we can easily measure the quality of the feature representations by reconstructing 3D assets and measuring their fidelity, and the assets are directly useful for downstream tasks like pose estimation or controlling generative models using methods like ControlNet (L. Zhang, Rao, & Agrawala, 2023). For

attributes like illumination and materials, however, it may be difficult to understand how 'good' the latent features are and what we can use them for.

One interesting application that has emerged in recent years is the potential use of these concepts to enhance the controllability of image-generative models. The current predominant condition used for image generation is text (so-called language prompts) due to its simplicity. However, text alone is often insufficient to fully convey what a designer imagines. Having control over these disentangled, and largely orthogonal 3D-aware attributes would allow fine-grained tuning over generated images, where the changes are often hard to describe via words.

Hence, this thesis is motivated by these two major gaps/research questions in currently available techniques of the general framework on learning 3D information from images (Figure 1.2): 1) How do we extract these 3D-aware information/features from the available datasets and 2) How do we learn to control downstream tasks such as generative models using these features?

1.2 Main Challenges

Different types of attributes and information require different foundational image models and different extraction methods. Therefore, we divide the attributes roughly into three categories: shape, continuous movement-based attributes, and materials. We define each of the categories as well as the corresponding main research challenges to extract and control attributes in each category for image generation/editing:

- Geometry – This refers to learning the shapes of each object from an image. Several works have tackled this ill-posed question by learning from Objaverse (Deitke et al., 2023; R. Liu et al., 2023; Shi et al., 2023; M. Liu et al., 2024).

Nevertheless, 2D datasets contain a much wider range of objects than these 3D datasets. The main challenge to explore is how to reconstruct full 3D shapes from just 2D image datasets. Such techniques could be beneficial to extending reconstruction to a wider range of unseen categories.

- Continuous Movement-Based Attributes – Many attributes, such as the motion of an object, movement of the light source, or displacement of the camera, are continuous attributes based on the movement of an element that requires a 3D scene to be modeled. These concepts are often more abstract – their effects are seen within the image, but are also often entangled with other attributes such as the geometry and materials of the object. In general, learning to extract and apply these motion changes to image generation and editing remains challenging, especially when many of these attributes can only be modeled by very few 3D objects available.

- Materials – This refers to the material properties of objects in the image. Explicitly extracting and transferring materials from one given exemplar to another image is difficult, as one has to perform full image decomposition to disentangle the geometry and lighting from the exemplar to obtain materials. Despite several efforts (Z. Li & Snavely, 2018; Zeng et al., 2024), accurate image decomposition remains an unsolved challenge. An interesting direction would be to explore whether it would be possible for large-image pretrained models to obtain and transfer this information implicitly, i.e., without having to explicitly model/remove attributes like geometry and illumination.

1.3 Key Contributions

Our thesis builds on the challenges described in Section 1.2. Specifically, each chapter is based on a research question from each described category. The key contributions for each part are as follows:

1.3.1 Discovering shapes from large image datasets without 3D supervision.

The ability to extract 3D shapes from arbitrary categories given an image is the bottleneck in many generative vision models to allow spatially-aware edits of images. Thus, we investigate whether we can push the extent of learning 3D from large 2D datasets where no 3D ground truths are available.

In Chapter 3, we propose 3DMiner, an end-to-end framework that discovers shapes from large-scale unannotated image datasets. We make a hypothesis that a very large image dataset must contain images representing the same shape, but varying in terms of backgrounds, pose, textures, lighting, etc. By using large pretrained feature extractors, we can filter and obtain good image clusters and discover coarse shapes for each cluster without the need for any 3D supervision.

1.3.2 Extracting and controlling motion-based 3D information (e.g., time-of-day illumination) in image generation.

The ability of text-to-image diffusion models to generate photorealistic images implies that their feature space implicitly contains knowledge regarding illumination, camera parameters, non-rigid shape changes, etc. In this stream of work, we investigate whether we can isolate these information with very little data.

In Chapter 4, we propose Continuous 3D Words, a new way to encode these 3D-aware attributes into continuous tokens, which can be used as sliders in conjunction with text during image generation and editing. By using few images rendered from rendering engines given one or very few meshes, we show that current text-to-image diffusion models (Rombach et al., 2021) can already disentangle and extract these high-level concepts to perform such a task.

1.3.3 Extracting and applying materials from one exemplar image to another.

Motivated by the powerful features in diffusion models which Continuous 3D Words shows, we investigate whether material properties are readily available within these features as well.

In Chapter 5, we present *ZeST*, a zero-shot, training-free approach to perform such image-to-image material transfer. By specifying geometry and illumination guidances from other sources, we can isolate the material information from general image encoders like CLIP (Radford et al., 2021). *ZeST* can capture and disentangle optical properties like reflection and transparency and apply them accordingly to the subject in the newly given image following its geometry and illumination cues.

1.4 List of Publications

Below we list the main publications contributing to each chapter. As a first author for all the publications, I am in charge of the majority of idea/design choices, code base, paper writing, experiments, diagrams, and visualizations. The project page for each work is linked at the beginning of every chapter.

- Chapter 3: "3DMiner: Discovering Shapes from Large-Scale Unannotated

Image Datasets". **Ta-Ying Cheng**, Matheus Gadelha, Soren Pirk, Thibault Groueix, Radomir Mech, Andrew Markham, Niki Trigoni. In IEEE/CVF International Conference on Computer Vision (ICCV), 2023.

- Chapter 4: "Learning Continuous 3D Words for Text-to-Image Generation". **Ta-Ying Cheng**, Matheus Gadelha, Thibault Groueix, Matthew Fisher, Radomir Mech, Andrew Markham, Niki Trigoni. In IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2024.
- Chapter 5: "ZeST: Zero-Shot Material Transfer from a Single Image". **Ta-Ying Cheng**, Parfull Sharma, Andrew Markham, Niki Trigoni, Varun Jampani. In European Conference on Computer Vision (ECCV), 2024.

2 | Literature Review

This literature review broadly covers the related works that are set as the foundations of the next three chapters. Specifically, we provide an overview of the works that motivate our problem setting, previous methods which our methods built upon, competing techniques, and where our work stands within the rapid advancements of 3D computer vision research. Figure 2.1 presents a taxonomy of the prior works as well as how the core contribution in each chapter contributes to the domain.

First, we describe the motivation of our work in detail by introducing the recent advancements in 2D and 3D datasets. We provide a comprehensive overview of the dataset sizes, diversity, and available annotations within image, video, and 3D domains.

Second, we dive into the works in 2D representation learning and 3D reconstruction methods which contribute to the core problem formulation of our work, 3DMiner, in Chapter 3. At a high level, we hope to utilize the strong priors of image representation to gain geometry features, all without the need for any 3D ground-truth data.

Finally, we discuss the advances in generative diffusion models from large image datasets, as it is the base foundation model for our work on Continuous 3D Words and on ZeST in Chapters 4 and 5. In particular, we look into works on 1) learning new concepts in diffusion models and 2) image editing in diffusion models, which form the baselines our works need to compare against.

A brief summary after each major section is provided to discuss where our contribution stands in this rapidly/changing research community.

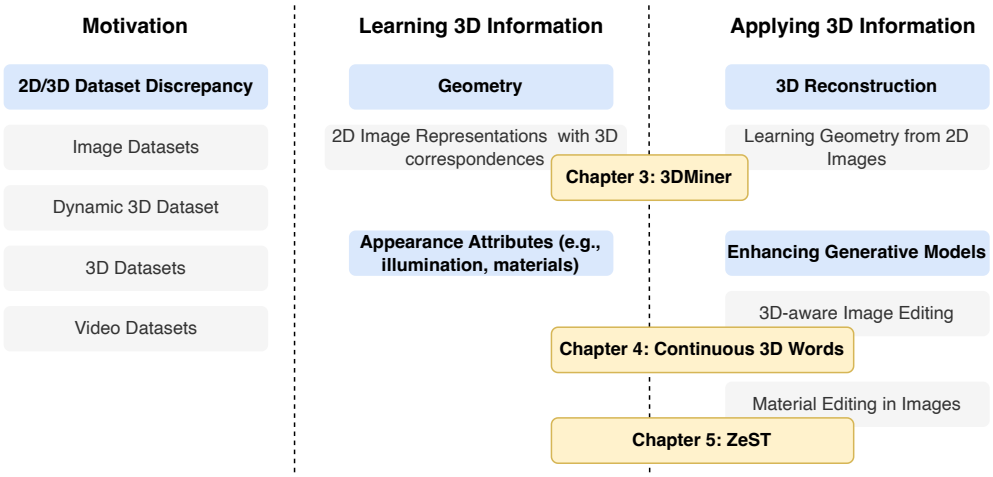

Figure 2.1: Taxonomy of related work and how our work is positioned. Chapter 3 combines the techniques in learning shapes from 2D image representations and implicit geometry methods to create an unsupervised 3D reconstruction method. Chapters 4 and 5 focus on a new domain of learning 3D information implicitly in the latent space, then building on top of current techniques in controlling generative models to provide new methods for image generation and editing.

2.1 Diversity of 2D and 3D Datasets

The core motivation of this thesis is built upon the lack of diversity in 3D datasets when compared against their 2D counterparts. The following section discusses the brief history of advancements in datasets within each domain (image, video, 3D) to highlight their differences.

2.1.1 The Advance in Image Datasets

Ever since the bloom of ImageNet in 2012 (Krizhevsky, Sutskever, & Hinton, 2012) and the drastic improvements gained in recognition tasks training from these large datasets (Simonyan & Zisserman, 2014; He, Zhang, Ren, & Sun, 2016; Ren, He, Girshick, & Sun, 2016; He, Gkioxari, Dollár, & Girshick, 2017), much work in the computer vision community has been committed to increasing the size of

image datasets with semantically meaningful labels, ranging from bounding boxes (T.-Y. Lin et al., 2014) to masks (T.-Y. Lin et al., 2014; Ravi et al., 2024).

The efforts in labeling and collection, accompanied by the blossoming internet-scale available images, soon led to big dataset efforts like WebImageText datasets, which sparked the era of contrastive language-image pretraining (CLIP (Radford et al., 2021)). These models are further used as filtering tools for further collection of even larger sets of image-text pairs. This led to the open-sourced LAION-400M and LAION-5B datasets (Schuhmann et al., 2022), which later fuelled the training of text-to-image generative models (Rombach et al., 2021).

The scale of this size inevitably incorporates images of not just real-world objects/scenes, but also paintings and images with various styles. The rich compositions accompanied by detailed texts enrich the understanding of models, both in discriminative and generative tasks, that subsequently enabled much fine-grained controls over them (Betker et al., 2023).

2.1.2 Video Datasets

Video datasets are also abundant as it is also easy to obtain videos via simple phone cameras. Some of the notable large video datasets include WebVid10M (Bain, Nagrani, Varol, & Zisserman, 2021) which contains 10 million datasets of video-text pairs scraped from stock footage sites, and Youtube-BB datasets (Real, Shlens, Mazzocchi, Pan, & Vanhoucke, 2017) which contain 240K images with 5.6 million bounding boxes.

The rich labels of videos significantly help with tasks like video captioning (B. Wang, Ma, Zhang, & Liu, 2018), segmentation (Tsai, Yang, & Black, 2016; Xu, Fu, Yang, & Lee, 2018), and subsequently generation (Blattmann et al., 2023; Guo et al., 2023).

2.1.3 3D Object Datasets

3D datasets, on the other hand, had a much slower start and growth.

After the development of ImageNet, several efforts have also been targeted towards 3D datasets. The most notable large datasets are ShapeNet and ModelNet40 (Chang et al., 2015; Z. Wu et al., 2015). ShapeNet comprises 3.3 million models with roughly 200k of them classified into various categories. A subset called ShapeNetCore is predominantly used as the benchmark for 3D data, as it contains 51K models from 55 categories. A subset of it contains part segmentations into several common categories such as tables and chairs. ModelNet40, on the other hand, contains roughly 12K CAD-generated objects from 40 categories.

The variety of the aforementioned 3D datasets, when compared to 2D image datasets, is significantly limited due to the difficulty in 3D data collection. The ease of taking photos anywhere ultimately creates a billion-scale database on the internet which most of the recent datasets (Schuhmann et al., 2022) came from. 3D data, on the other hand, requires additional sensors for either depth estimations or pose estimations to properly reconstruct the scene. Ultimately, the objects created from these datasets are single-object and synthetic, inevitably presenting a domain gap between them and real-world 3D objects.

Several works followed ShapeNet and ModelNet and went beyond synthetic datasets and renderings with in-the-wild images. Pix3D contains 10k examples of objects from 9 categories in from Ikea dataset, accompanied by in-the-world images taken under various environments with known camera poses (Sun et al., 2018). ScanObjectNN obtains real scans of objects with backgrounds to mimic real scenes (Uy, Pham, Hua, Nguyen, & Yeung, 2019). CO3D comprises 5625 point-cloud-annotated videos with camera poses (Reizenstein et al., 2021). Om-

niObject3D contains 6k richly annotated data with videos, poses, and ground truths from 190 categories (T. Wu et al., 2023).

Recently, the collective effort of game/graphic designers brought about a large leap in the size and variety of 3D assets. This led to the development of Objaverse and subsequently Objaverse-XL (Deitke et al., 2023, 2024), which comprises 10 million 3D objects. Nevertheless, while the quantity of data is approaching 2D datasets (each 3D object can be converted to multiple views), the diversity of these objects is much more limited. The majority of data are single object with similar styles. This limits the semantic context a model can learn from.

2.1.4 Dynamic 3D Datasets

A subset of Objaverse data also contains motions. Some efforts have been made to filter out the high-quality data that can be used to learn multi-view video generation. These led to the dataset creations of VividZoo, Animate3D, and Objavers-Dy (B. Li et al., 2024; Jiang et al., 2024; Y. Xie, Yao, Voleti, Jiang, & Jampani, 2024). However, while all of them have been carefully filtered, they are all rendered in white backgrounds with a single moving object in front.

On the other hand, there are dynamic scene-level datasets that are created by simulation engines. Examples of these include Kubric-4D and ParallelDomain-4D (Van Hoorick et al., 2024). However, they are both domain-specific (Kubric-4D only has rigid object-falling animations, whereas ParallelDomain-4D only has traffic scenes). Scene-level dynamic 3D datasets capturing non-rigid objects are still somewhat limited today.

2.1.5 Summary

Overall, the diversity of 3D datasets is noticeably inferior compared to 2D image and video datasets. Hence, a way to distill 3D information such as geometry, motion, and appearances from 2D datasets is an important research question to bridge 2D-3D understanding.

2.2 Extracting Shape Information from 2D Image Datasets

2.2.1 Image Representation Learning and Its 3D Properties

Learning useful features/representations from images agnostic to the downstream tasks has been something long sought for in the image community. In the deep learning era, it gained popularity owing to the introduction of contrastive learning (Oord, Li, & Vinyals, 2018). Contrastive learning aims to encourage similarity between representations of similar samples and push away pairs that are not. Specifically, given a set of samples $X = \{x_1, ..., x_N\}$ where $x_i, x_j \in X, i \neq j$ forms a positive pair and the rest are negative pairs, the contrastive loss is defined as:

$$\mathcal{L}_{infoNCE} = -\mathbb{E}_X[\log \frac{sim(x_i, x_j)}{\sum_{x_k \in x, k \neq i} sim(x_i, x_k)}], \tag{2.1}$$

where $sim(\cdot)$ is the similarity function. Chen et al. later proposed SimCLR that applied Equation (2.1) on heavily augmented pairs of images to train in a self-supervised manner. They empirically showed that the created embedding space, while trained without any labels, can achieve stunning accuracies in recognition tasks such as image classifications by attaching a nearest neighbour/simple

neural network algorithm afterward. In other words, the contrastive criterion self-supervised the network in learning meaningful features for distinguishing classes of instances.

A plethora of work soon followed. Some notable arts include Momentum Contrast (MoCo) which introduces a dynamic dictionary for contrastive learning (He, Fan, Wu, Xie, & Girshick, 2020), Nearest-Neighbour Contrastive Learning (NNCLR) which samples nearest neighbours and treats them as positive pairs (Dwibedi, Aytar, Tompson, Sermanet, & Zisserman, 2021), and Decoupled Contrastive Learning (DCL) that omits the positive term in the denominator to boost results on smaller batch sizes (C.-H. Yeh et al., 2022). Non-contrastive self-supervised methods also emerged during this era. Bootstrap Your Own Latent (BYOL) relies on two neural networks learning from each other and surprisingly outperformed several contrastive methods (Grill et al., 2020). Masked Autoencoder learns image representations by masking out learning to reconstruct missing patches (He et al., 2022).

In 2021, DINO-ViT applied a self-supervised knowledge distillation method on vision transformers, and showed that such methods learn not only class-specific representations but also fine-grained features that can be directly applied for object segmentations (Caron et al., 2021). Amir et al. further explored the capabilities and found out that dense correspondences, part segmentations, and co-segmentations can also be achieved with pretrained DINO-ViTs (Amir, Gandelsman, Bagon, & Dekel, 2021). More recently, improvements on the dataset and design choices of the DINO architecture were further explored, leading to the introduction of DINOv2 (Oquab et al., 2023). The model can now perform part correspondences even for objects that are semantically different (e.g., matching a bird's wing with a plane's wing). These advances are significant, as they could potentially be useful not only for traditional recognition tasks, but also for more challenging problems

such as reconstructing the geometry of objects given multiple images from different viewpoints.

2.2.2 3D Geometry from Images

Extracting important features to inflate visual data into 3D geometry has been a long-standing problem to bridge 2D and 3D scene understanding. Numerous deep learning-based approaches utilise a 2D encoder followed by a 3D decoder (Choy, Xu, Gwak, Chen, & Savarese, 2016; H. Xie, Yao, Sun, Zhou, & Zhang, 2019; H. Xie, Yao, Zhang, Zhou, & Sun, 2020), and thus we can view the encoder as a representation learning procedure to distill geometry-aware features. In addition to the shape, several works also simultaneously predict the location and pose of the object (Gkioxari, Malik, & Johnson, 2019; Kuo, Angelova, Lin, & Dai, 2020). One notable example is AutoRF (Müller et al., 2022), which captures an object-centric representation, encompassing disentangled information regarding shape, appearance, and pose of the targeted image. These object-centric information are then decoded to the target views. Many methods also introduce category-specific shape priors, where a template deduced from the training set of 3D shapes is also used as input to the encoder to better learn the latent representation (S. Yang, Xu, Xie, Perry, & Xia, 2021; Wallace & Hariharan, 2019; Michalkiewicz et al., 2020; Cheng, Yang, Trigoni, Chen, & Liu, 2022). For example, CodeNeRF (Jang & Agapito, 2021) presents a pipeline of learning to encode general class categories during training, then decoding novel views for unseen specific instances during inference.

The effect of priors is particularly significant under few-shot reconstruction scenarios (Wallace & Hariharan, 2019; Michalkiewicz et al., 2020). However, these methods would require ground-truth 3D objects for reference, which is hardly

scalable and transferrable to novel categories.

A series of self-supervised methods are also introduced to relax the aforementioned constraints. Angjoo et al. proposed a category-specific mesh reconstruction where a set of consistent keypoints within a class is used to predict shape, camera, and texture given an image (Kanazawa, Tulsiani, Efros, & Malik, 2018). The more recent techniques such as SDF-SRN (C.-H. Lin, Wang, & Lucey, 2020) and TARS-3D (Duggal & Pathak, 2022) learn signed distance functions implicitly for single-view reconstruction. In particular, TARS-3D seems to generate the most accurate shapes on complex topologies like chairs, but it still requires a set of synthetic datasets (e.g., ShapeNet) to facilitate the training.

Hu et al. further proposed SMR, a self-supervised method comprising a 2D rendering and 3D comparison loss. Their method requires only images and their corresponding silhouettes to learn the 3D representations (T. Hu, Wang, Xu, Liu, & Jia, 2021). Most recently, Monnier et al. proposed a progressive conditioning strategy to even eliminate the need for silhouettes in their most recent method termed Unicorn (Monnier, Fisher, Efros, & Aubry, 2022). Nevertheless, despite the gradual improvements in accuracies and removal of label requirements, these methods are still far from generalisable across different datasets. All the comparisons mainly exist on synthesized datasets (e.g., ShapeNet (Chang et al., 2015)) or very constrained real-world datasets (e.g., CUB-200 (Wah, Branson, Welinder, Perona, & Belongie, 2011)) where the background and foreground are of very different data distribution. Accurate methods on more generalised datasets such as Pix3D (Sun et al., 2018) are still in vain, not to mention the inability to extend to even larger datasets such as LAION-5B (Schuhmann et al., 2022).

2.2.3 Position of Our Work in Literature

Unlike prior methods that either require additional 3D labels (e.g., keypoints) or can lack generalisation to real-world datasets, our work in Chapter 3 aims to provide a framework that can work on any large unannotated 3D dataset. By combining the best of 2D image representation and multi-view reconstruction techniques, we create an end-to-end pipeline that mines and filters 3D shapes from just large pretrained models learnt from 2D datasets and labels.

2.3 Applying 3D-Awareness to Generative Models

In addition to learning shapes from image-pretrained models, we are also interested in extracting 3D-aware concepts and controlling them for downstream tasks such as generation. The following section aims to provide an overview of the current literature on controllability of diffusion-based generative models and their 3D awareness, which forms the foundation of the works we present in Chapters 4 and 5.

2.3.1 Overview of Diffusion Models

Diffusion models for images are generative models that are trained to denoise an example from Gaussian distribution to a sample in the image data distribution (Ho, Jain, & Abbeel, 2020). The objective of this model, termed S_θ is to learn to predict the noise at a given timestep and a noised image by minimizing the following objective:

$$\mathbb{E}_{x,t,\epsilon}\left[\|S_\theta(\alpha_t x + \sigma_t \epsilon, t) - x\|_2^2\right], \tag{2.2}$$

where x is the original image sample, $\alpha_t x + \sigma_t \epsilon$ is the noised image being controlled

by the noise schedule at time t.

The setup pushed the quality and generalizability of image generation beyond GANs (Goodfellow et al., 2020). This is predominantly a result of the denoising process being easier to optimize than the combat of the generative and discriminative model in the GAN setup.

Stable Diffusion further popularized this generation using diffusion models by employing memory-efficient models through latent-space diffusion (Rombach et al., 2021). This allows images to be generated at much higher resolution with much lower costs. Today, there exist numerous variations of diffusion-based generative models generating highly realistic and aesthetic images (Esser et al., 2024; Betker et al., 2023).

2.3.2 Controlling Image Generation and Editing

With the great improvement in image quality, the vision community has introduced a wide range of modalities that can be used as conditions to control the image generation process. The most dominant condition ought to be text.

Works such as DALLE (Ramesh, Dhariwal, Nichol, Chu, & Chen, 2022; Ramesh et al., 2021; Betker et al., 2023) and Imagen (Saharia et al., 2022) used large-scale text-image datasets and strong language understandings from pretrained LLMs (Brown et al., 2020; Raffel et al., 2020; Devlin, Chang, Lee, & Toutanova, 2018) to guide the generation process. In particular, the text features are injected as conditions into the diffusion model for the noise prediction.

Other works built on top of these models by adding other forms of conditioning (L. Zhang et al., 2023; Mou et al., 2023). Highly relevant to our work is ControlNet (L. Zhang et al., 2023), which proposes a general pipeline with zero-

convolutions for conditioning on text and image data (e.g., depth maps, canny maps, sketches). Despite their impressive image quality, their control is based entirely on hints provided in the pixel space. It is not clear how to use these models to control other attributes of images that are more abstract, like illumination or object orientation.

Other than controlling image generation, another stream of work focused on editing existing images. The majority of work in this space explores how to perform image edits using textual instructions. Given a text-generated image, they demonstrate how the user can edit the image by amending the prompt, yet still preserve some aspects of the original image (Parmar et al., 2023; Hertz et al., 2022; Brooks, Holynski, & Efros, 2023). While convenient, these approaches have the same issue with the aforementioned text-to-image generation techniques: they do not allow control over concepts that are very difficult to be described by texts.

Recently, as the amount of 3D data available significantly increased, Liu et al. (R. Liu et al., 2023) introduced Zero-1-to-3, a diffusion model trained on various viewpoints of 3D rendering that enables viewpoint editing given an image of a single object. Similarly, works like DreamSparse (Yoo, Guo, Matsuo, & Gu, 2023) also employ diffusion models to synthesize novel views on open-set categories. Nevertheless, these techniques are focused only on object orientation and rely on vast 3D shape datasets and are not generalisable to many continuous concepts, such as e.g. illumination, wing pose, dolly zoom, that can be directly applied in text-to-image scenarios.

2.3.3 Learning New Concepts on Diffusion Models

With diffusion models being trained on unforeseen quantities in images and texts, a stream of work focused on distilling specific concepts with very few data samples.

For example, given a small set of images representing one particular object instance, textual inversion learns a new word embedding to describe the object, such that the word can be applied with new text prompts for image generation (Gal et al., 2022). NETI (Alaluf, Richardson, Metzer, & Cohen-Or, 2023) extended the word embedding to a time-space conditioned neural mapper for better generation while preserving quality. Similarly, Dreambooth (Ruiz et al., 2023) aims to achieve the same goal, but by using a repurposed token rarely used in text and finetuning the entire diffusion model with an additional constraint to prevent generative loss. There are numerous subsequent works showing improvements in finetuning different layers/weights and by improving the training strategy (Kumari, Zhang, Zhang, Shechtman, & Zhu, 2023; Han et al., 2023; Z. Liu et al., 2023).

Despite the advances in adding new personalized entities to existing models, few works focus on learning general concepts that can be applied to a variety of scenarios. ViewNETI (Burgess, Wang, & Yeung, 2023) is the first to learn viewpoints as a concept, but we hypothesise that the 3D awareness of large text-to-image diffusion models goes far beyond merely viewpoints, allowing us to associate and even create interactions with multiple 3D-aware concepts like illumination, pose and camera parameters *at the same time*.

2.3.4 Material Editing in Images

While most generative works focus on semantic-based editing of images, another stream of work focuses on editing material properties of objects within images. It is an important subset of problems that shares the same challenge of being very difficult to control via text. In addition, the understanding of materials often requires explicit extraction so that the material can be applied to other objects.

Several works focused on the extraction of materials from images (T. Y. Wang,

Ritschel, & Mitra, 2018; Meka et al., 2018), while other works have introduced material editing within an image with depth estimations (Khan, Reinhard, Fleming, & Bülthoff, 2006). More recently, works have utilized generative models for perceptual material editing (Subias & Lagunas, 2023; Delanoy, Lagunas, Condor, Gutierrez, & Masia, 2022). With the recent advance in diffusion models, (Sharma, Jampani, et al., 2023) proposed shader-based editing by rendering a synthetic image dataset of materials with various parametric properties (e.g., metallic, transparency, roughness). Generative models have also been employed for explicit material learning (Lopes, Pizzati, & de Charette, 2023) and subsequently 3D mesh texturing (Y.-Y. Yeh et al., 2024; D. Z. Chen, Siddiqui, Lee, Tulyakov, & Nießner, 2023; Richardson, Metzer, Alaluf, Giryes, & Cohen-Or, 2023; T. Cao, Kreis, Fidler, Sharp, & Yin, 2023).

2.3.5 Position of Our Work in Literature

The main challenge in prior literature is the lack of fine-grained control over many 3D-aware attributes. Chapters 4 and 5 tackle different aspects within this domain gap.

In Chapter 4, we focus on the particular challenge of editing continuous movement-based attributes in an image, where an element in the 3D scene (e.g., time-of-day illumination [] , orientation [ ]) moves, causing a change in the image. We derive a way to present these elements as special continuous words that act like sliders. These words can then be used jointly with other texts for fine-grained control over image generation and editing.

In Chapter 5, we take one step further and explore the 3D awareness of pretrained 2D diffusion models on understanding and transferring materials. We show that a meticulously designed pipeline can allow us to disentangle and perform image-to-

image material transfer, all without any further training.

3 | Discovering Shapes from Large Unannotated Image Collections

The main content of this chapter is published and presented in ICCV 2023.

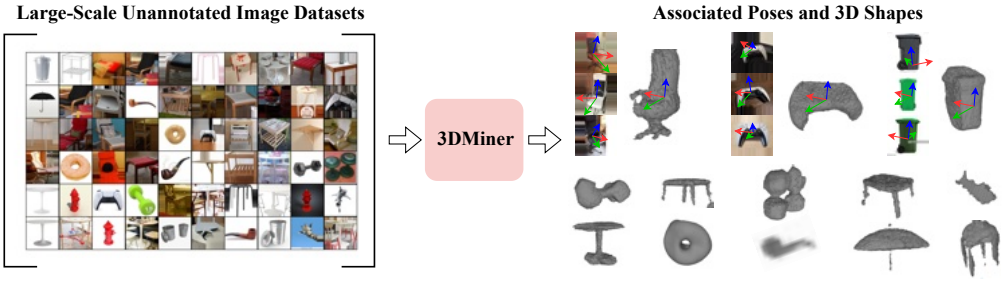

Figure 3.1: **Overview**. We present 3DMiner, a scalable framework designed to obtain associating poses and reconstruct shapes from *diverse and realistic* sets of images *without any 3D data, pose annotation, camera information, or keypoints*.

In this chapter, we explore the problem of extracting geometry from 2D image datasets without any supervision. Specifically, we present 3DMiner – a pipeline for mining 3D shapes from challenging large-scale unannotated image collections. Unlike other unsupervised 3D reconstruction methods, we assume that, within a large-enough dataset, there must exist images of objects with similar shapes but varying backgrounds, textures, and viewpoints. Our approach leverages the recent advances in learning self-supervised image representations to cluster images with geometrically similar shapes and find common image correspondences between them. We then exploit these correspondences to obtain rough camera estimates as initialization for bundle-adjustment. Finally, for every image cluster, we apply a progressive bundle-adjusting reconstruction method to learn a neural occupancy field representing the underlying shape. We show that this procedure is robust to several types of errors introduced in previous steps (e.g., wrong camera poses,

images containing dissimilar shapes, etc.), allowing us to obtain shape and pose annotations for images in-the-wild. When using images from Pix3D chairs, our method is capable of producing significantly better results than state-of-the-art unsupervised 3D reconstruction techniques, both quantitatively and qualitatively. Furthermore, we show how 3DMiner can be applied to in-the-wild data by reconstructing shapes present in images from the LAION-5B dataset. Project Page: https://ttchengab.github.io/3dminerOfficial.

3.1 Introduction

Learning-based systems that try to reason about 3D geometry from images suffer from a fundamental limitation: the amount of available 3D data. Despite recent advances in capturing the world tridimensionally, the biggest image datasets still contain many orders of magnitude more data points than their 3D counterparts. In practice, the sheer quantity of images ends up capturing a much richer visual vocabulary – different textures, illuminations, shapes, environments, types of objects and relationships between them. Therefore, developing techniques that can leverage this information is the key to general, high-performing 3D reconstruction algorithms. Unfortunately, the abundance and variety of visual data in image datasets leads to great complexity when trying to extract 3D information. Consider a very simple dataset consisting of multiple images of the same object in the same environment taken from different camera viewpoints. Extracting 3D information is not entirely trivial, but potentially doable through stucture-from-motion approaches. However, if we increase the complexity of this dataset by adding images of the object in different environments, different materials and with slight shape variations, structure-from-motion techniques will fail. The complexity only increases when we consider all the possible image permutations that one can find online: a myriad

of object types, occluded objects, partial observations, non-photorealistic imagery and so on. How can we extract 3D information from such complicated image datasets?

Various image-to-3D approaches have tried to tame the visual complexity in big image datasets. They usually employ different amounts of manual image annotations; *i.e.,* object poses, masks, keypoints, part segmentations, and so on. These techniques use models that are trained to disentangle 3D geometry from various other factors while trying to reconstruct the original image and its annotations. Due to the ill-posed nature of the single-view reconstruction problem, training these models is very hard and multiple regularizations are necessary. When presented with more challenging datasets, with real images, even if they only depict a single object type (e.g., Pix3D chairs), the best models fail to produce reasonable results.

In this work we aim to extract 3D shapes from image datasets in a completely different way. We call our approach 3DMiner. Given a very large set of images, our initial goal is to separate them in groups containing similar shapes. Within these groups, we estimate robust pairwise image correspondences that will give us a good idea about the relative pose of the objects in the images. Using this information, we can estimate the underlying 3D geometry in every image group, effectively treating the single-view reconstruction problem as a noisy multi-view one. Unfortunately, this is not a straightforward structure-from-motion setup – within the same group, the objects are similar but not exactly the same; they have different colors, backgrounds, and even slightly different geometry. To circumvent this issue, we adopt modern reconstruction techniques based on neural fields. These representations give us the ability to train parametric occupancy fields through gradient descent while refining camera poses, intrinsics, and more importantly, giving us a proxy for the quality of the recovered 3D shape – the image reconstruction loss. Ultimately, the entire pipeline provides association between

the shape and poses across images in-the-wild for arbitrary categories – a task for which many datasets rely on manual annotations.

Instead of having a single hard-to-train model extracting shapes from images, we opt for dissecting the problem and breaking it in pieces that can be tackled with well-studied tools. At the heart of our approach are recent advances in learning representations from images in an unsupervised manner. Those techniques allow us to identify image associations and to find common features more conveniently, establishing robust correspondences and ignoring nuisance factors like different backgrounds, illumination and small shape variations. As a notable example, the recent DINO-ViT, trained through self-supervision on ImageNet data, has shown remarkable ability to distinguish foregrounds, perform part segmentation, and generate common keypoints. More importantly: further improvements in image representation learning can be *immediately* incorporated into our method – no model needs to be retrained or fine-tuned. The same can also be applied to advances in neural fields. Once better methods for optimizing occupancy fields are developed, they can be directly plugged into our pipeline.

We refer to our method as unsupervised, meaning that it does not require 3D data to perform 3D reconstruction. Thus, using other features/outputs from models trained without 3D data do not alter the unsupervised 3D reconstruction setup. We do, however, restrain from using keypoints/pose estimators (e.g., bird keypoints) as they limit our approach to category-specific conventions.

We demonstrate the capabilities of our approach in two empirical studies using Pix3D and LAION-5B. For Pix3D, we use our 3D mining pipeline as a way of generating 3D for each image in the dataset and directly compare it against state-of-the-art single-view reconstruction approaches that do not use 3D information during training. Our experiments show that, differently from the other approaches, 3DMiner is capable of generating reasonable 3D shapes and displays a relative

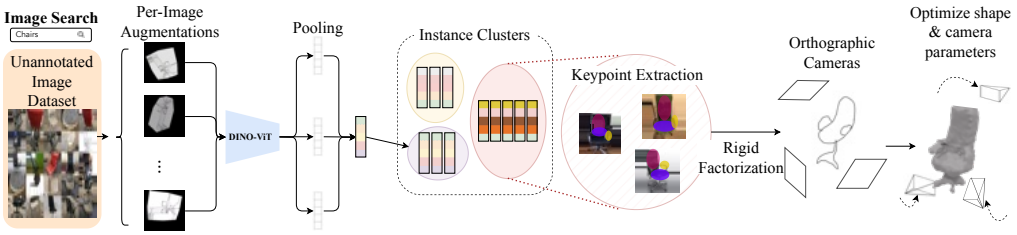

Figure 3.2: **3DMiner pipeline.** Our method starts by grouping images that depict similar 3D shapes, regardless of the texture of the shape, the camera view-point or the background. To do so, we perturb each image with various transformations (e.g. color jittering, perspective and rotation) and we pool their DINO-ViT features to create a robust image embedding. We cluster images by running agglomerative clustering on the embeddings. Within each cluster, we find key point correspondences using dense DINO-ViT features. We feed those corresponding keypoints to a Structure from Motion algorithm (rigid factorization) to get coarse orthographic camera estimations. Finally, we jointly refine the camera parameters and learn an occupancy field to get the final shape.

F-score improvement of **80%** over the state-of-the-art. In order to showcase the versatility of our approach we also ran 3DMiner on subsets of images from LAION-5B gathered using various short text prompts. Despite the challenges presented by the diversity in this dataset, 3DMiner is still capable of finding reasonable 3D representations across multiple categories.

In summary, our contributions are: *(i)* a new problem formulation on mining 3D shapes from large-scale web-retrieved images without any priors or annotations; *(ii)* an end-to-end pipeline to cluster, estimate pose, and generate neural 3D representations from unannotated image datasets without any 3D ground truths; *(iii)* a detailed empirical study to showcase the superiority of our method on challenging datasets and robustness under categories where no previous ground-truth reconstructions exist.

3.2 Related Work

Multi-view reconstruction, or *Structure from Motion (SfM)*, assumes a set of images of a stationary scene and jointly reconstructs a 3D point cloud of the scene along with the camera parameters of each image. Since the seminal work of Longuet-Higgins (Longuet-Higgins, 1987), SfM has been extensively researched (Tomasi & Kanade, 1992; Triggs, 1996; Snavely, Seitz, & Szeliski, 2006; Agarwal, Snavely, Simon, Sietz, & Szeliski, 2009). Before optimization, the camera parameters are typically initialized by either applying RANSAC (Bolles & Fischler, 1981) or matrix factorization on pairs of keypoints. Classical approaches to keypoint matching detect candidate matches in each image and embed them into a descriptive feature space. More recently, classical descriptors (LoweDavid, 2004) have been outperformed by learned approaches (DeTone, Malisiewicz, & Rabinovich, 2018; Sarlin, DeTone, Malisiewicz, & Rabinovich, 2020). Interestingly, features from self-supervised approaches, like DINO-ViT (Caron et al., 2021), have proven to be robust in a wide range of downstream tasks, including keypoint matching (Amir et al., 2021).

More recently, Mildenhall *et al.* (Mildenhall et al., 2020) demonstrated photo-realistic performance on the task of novel-view synthesis using differentiable volume rendering of neural radiance fields. NeRFs (Mildenhall et al., 2020) initially assumed multiple views of the same stationary scene with known camera poses and several extensions are relevant to our work. NeRF-W (Martin-Brualla et al., 2021) tries to generalize NeRF to more diverse conditions, in particular transient occluders and illumination changes. Several approaches address 3D reconstruction instead of novel view synthesis and optimize directly sign-distance fields (J. Zhang, Yao, & Quan, 2021; Yariv et al., 2020; P. Wang et al., 2021; Yariv, Gu, Kasten, & Lipman, 2021) or occupancy fields (Niemeyer, Mescheder,

Oechsle, & Geiger, 2020; Oechsle, Peng, & Geiger, 2021). Several works also tackle the problem of multi-view reconstruction with no camera parameters known. NeRF−− (Z. Wang, Wu, Xie, Chen, & Prisacariu, 2021) and BARF (C.-H. Lin, Ma, Torralba, & Lucey, 2021) jointly optimize the camera parameters and the radiance field during training.

However, SfM approaches, NeRF, and its variants assume photometric consistency between the different views of the scene. On the contrary, our approach, 3DMiner, does not rely on this assumption and reconstructs 3D shapes from image clusters containing roughly the same geometric shape with different textures and backgrounds. We draw inspiration from the classical SfM pipeline: we use DINO-ViT (Caron et al., 2021) features to estimate corresponding keypoints across images with different textures and backgrounds, and use those keypoints to estimate coarse camera poses. Similar to BARF (C.-H. Lin et al., 2021) and space carving (Kutulakos & Seitz, 1999), we then jointly reconstruct an occupancy field and perform bundle-adjustments, using a silhouette loss. Silhouettes can be estimated using the saliency from DINO-ViT features and further refined using saliency segmentation models.

Learning shapes with 3D data. Several approaches leverage existing datasets of 3D shapes, *e.g.,* ShapeNet (Chang et al., 2015), ModelNet (Z. Wu et al., 2015), Pix3D (Sun et al., 2018), to learn 3D reconstructions. Their most distinctive feature is usually the choice of 3D representation. 3D geometry can be represented via voxel grids (Choy et al., 2016; H. Xie et al., 2020; Cheng et al., 2022), point clouds (Fan, Su, & Guibas, 2017; Mandikal, Navaneet, Agarwal, & Babu, 2018), meshes (Groueix, Fisher, Kim, Russell, & Aubry, 2018; N. Wang et al., 2018; Ganapathi-Subramanian et al., 2018), or implicit functions (Mescheder, Oechsle, Niemeyer, Nowozin, & Geiger, 2019; Z. Chen & Zhang, 2019; Park, Florence, Straub, Newcombe, & Lovegrove, 2019; Yu, Ye, Tancik, & Kanazawa, 2021). A

common challenge for this type of approach is to generalize beyond the limited number of categories they are trained on. On the contrary, 3DMiner seeks to leverage large-scale in-the-wild 2D datasets for their potential to cover a wider range of objects and not limited to specific categories.

Learning shapes without 3D data. A large corpus of work in the *Single-View Reconstruction (SVR)* community focuses on directly learning 3D from 2D images, without any 3D annotation. Several approaches supervise SVR with multiple views of the same scenes using differentiable volumetric rendering (Tulsiani, Zhou, Efros, & Malik, 2017; Yan, Yang, Yumer, Guo, & Lee, 2016; Jimenez Rezende et al., 2016; Niemeyer et al., 2020) or differentiable mesh renderers (Kato, Ushiku, & Harada, 2018; S. Liu, Li, Chen, & Li, 2019; W. Chen et al., 2019). Notably, (Tulsiani, Efros, & Malik, 2018; Insafutdinov & Dosovitskiy, 2018) do not assume known camera poses but estimate them jointly.

To overcome the ill-posed nature of the problem, several forms of priors have been explored, including keypoints correspondences (Kar, Tulsiani, Carreira, & Malik, 2015; Vicente, Carreira, Agapito, & Batista, 2014; Tulsiani et al., 2017; Kanazawa et al., 2018; Duggal & Pathak, 2022; C.-H. Lin et al., 2020; Henderson, Tsiminaki, & Lampert, 2020), silhouette losses (Kanazawa et al., 2018; Goel, Kanazawa, , & Malik, 2020; Tulsiani, Kulkarni, & Gupta, 2020; T. Hu et al., 2021; S. Wu et al., 2021; Y. Ye, Tulsiani, & Gupta, 2021), shape templates (Goel et al., 2020; Tulsiani et al., 2020), different forms of symmetry (Henderson et al., 2020; Goel et al., 2020; Kanazawa et al., 2018; X. Li et al., 2020; T. Hu et al., 2021) including rotation symmetries (S. Wu et al., 2021). Several approaches leverage off-the-shelf 2D networks (X. Li et al., 2020; Y. Ye et al., 2021) or generative adversarial techniques (Henzler, Mitra, & Ritschel, 2019; Gadelha, Maji, & Wang, 2017; Kato & Harada, 2019; Pavllo, Spinks, Hofmann, Moens, & Lucchi, 2020; Y. Ye et al., 2021). Particularly relevant to us are Unicorn (Monnier et al., 2022)

and SMR (T. Hu et al., 2021). In SMR, Hu *et al.* (T. Hu et al., 2021) use self-supervised learning to learn 3D from 2D images. Their method requires only images and their corresponding silhouettes. In Unicorn, Monnier *et al.* (Monnier et al., 2022) propose a progressive conditioning strategy, only assuming that images in the training set belong to the same category. We compare the performances of 3DMiner against these recent techniques. Note however, that while SMR and Unicorn tackle SVR, 3DMiner is not intended to be used for SVR but rather to automatically mine 3D content from large 2D collections of images.

Despite the gradual improvements in accuracy and the removal of label requirements, all of these methods are still not capable of yielding good results in more challenging datasets. Most comparisons mainly exist on synthetic data (Chang et al., 2015) or very constrained real-world images (Wah et al., 2011), where the background and foreground are very distinct and not many different shapes can be found. Previous approaches cannot handle data such as Pix3D (Sun et al., 2018) (unless trained with additional datasets), not to mention in-the-wild datasets such as LAION-5B (Schuhmann et al., 2022). The challenge comes from the fact that training the network to generate reasonable 3D geometry by using reprojection losses is very hard – it requires a lot of regularizations that lead to overly smooth geometry, otherwise yielding degenerate solutions. In contrast, our approach adopts a pipeline where images are grouped by shape similarity and the reconstruction in each group happens independently. This endows 3DMiner with the ability to not only tackle more challenging data but also leads to a system that readily benefits from progress in relevant subproblems (*e.g.,* image representation learning, landmark detection, pose estimation, neural field reconstruction) without requiring any retraining or fine-tuning.

			Threshold					
			0.5		0.4		0.3	
	CD ↓	**F1** ↑	**CD** ↓	**F1** ↑	**CD** ↓	**F1** ↑	**CD** ↓	**F1** ↑
SMR (T. Hu et al., 2021)	0.192	0.130	0.189	0.131	0.188	0.132	0.267	0.110
Unicorn (Monnier et al., 2022)	0.263	0.102	0.259	0.106	0.266	0.105	0.154	0.160
3DMiner (Ours)	**0.130**	**0.234**	**0.125**	**0.244**	**0.116**	**0.263**	**0.095**	**0.307**

Table 3.1: **Comparisons on Pix3D Chairs.** We align the meshes to their ground truths via Coherent Point Drift (Myronenko & Song, 2010) and compute the Chamfer Distance (CD) and F1 Score (F1). We select three subsets of images using a threshold on the reprojection error within each cluster. On the full Pix3D chairs (left two columns), 3DMiner improves by 33% the CD and 10 points the F1, and the performance increases significantly when we use our selection criterion (see Section 3.4.2 for more discussion).

3.3 Method

Overview. Our goal is to mine 3D shapes from large-scale in-the-wild image datasets. We assume that, given a large-enough dataset, there must exist several images of very similar shapes once ignored the differences in terms of backgrounds, textures, viewpoints, and lightning conditions. When the images containing similar shapes are grouped, we extract pairwise image correspondences to estimate orthographic camera poses for each image. Finally, this information will be used in a neural occupancy field optimization procedure that recovers shape and refines perspective camera poses. Figure 3.2 shows an overview of our method. In the following subsections we describe each step of this pipeline in detail.

3.3.1 Clustering Similar Shapes

We aim to find clusters of similar shapes, irrespective of their texture, background and illumination conditions. Caron *et al.* (Caron et al., 2021) showed that Vision Transformers (ViT) trained with self-supervision learned powerful features for classification tasks as well as semantic segmentation tasks. We take advantage of these models to perform a clustering of images containing similar shapes. Contrary

to 3D reconstruction approaches trained with 3D supervision on selected object categories, DINO-ViT has been trained on ImageNet (Russakovsky et al., 2015) and therefore is more robust when applied to in-the-wild imagery.

To make the clustering invariant to texture, background and the camera viewpoint, we propose a simple augmentation method. Formally, given an image I, we obtain a set of augmented DINO-ViT features $S_I = \{f(A_1(I)), ..., f(A_n(I))\}$, where A_i refers to a random set of augmentations involving color jittering, image rotation, and perspective transformation, and $f(\cdot)$ is the pretrained DINO-ViT outputing a per-image global feature. The final feature of the image z_I is then computed as:

$$z_I = [\max(S_I), \text{mean}(S_I), \min(S_I)]. \tag{3.1}$$

Color jittering helps to make the resulting feature more invariant to texture and illumination conditions. Rotation and perspective transformations mimic a change of viewpoints in 3D. These augmentations encourage the features to be dependent on the geometry itself.

Using the feature z_I from Equation 3.1, we perform a bottom-up agglomerative clustering. Section 3.4.4 presents an analysis of our augmentation scheme on the Pix3D datase and Figure 3.5 shows examples of clusters from the LAION-5B dataset.

3.3.2 Coarse Orthographic Pose Estimation

At this point, we assume we have a cluster of unannotated images of the same geometric shape and seek to estimate a coarse camera pose for each image in the cluster. Over a decade ago, Marques et al. proposed a matrix factorization technique to estimate orthographic poses from a sequence of images and corresponding keypoints (Marques & Costeira, 2009). Their method is robust to some amount of

noise in the keypoints and does not require that all images contain all keypoints. Perhaps for this reason it has been widely adopted in datasets such as CUB-200 and Pascal VOC where manually labelled keypoints are available (Kanazawa et al., 2018; Vicente et al., 2014; Kar et al., 2015). Thus, we will leverage this classical technique to estimate initial camera poses, but, differently from previous approaches, our keypoint correspondences are established using image representations learned through self-supervision. More specifically, we extract image features using the DINO-ViT model (same used for clustering).

Within a cluster, we adopt an approach inspired by part-segmentation method from (Amir et al., 2021). We extract features at different layers of the ViT for all spatial locations in all images, run k-means on this set of features, and select segments that are salients and common to most images using a voting strategy. We compute the bounding boxes of each segment in each image, and use its center as a keypoint. We show in the supplementary material a visualization of the estimated keypoints for multiple clusters.

Equipped with estimated keypoints within a cluster, we then perform rigid factorisation through SVD and Stiefel manifold projections following (Marques & Costeira, 2009) to obtain an orthographic camera for every image I in the format:

$$p_{2d} = M \cdot p_{3d} + t, \tag{3.2}$$

where $M \in \mathbb{R}^{2\times3}$ and $t \in \mathbb{R}^{2\times1}$ are the orthographic motion and translation matrix to project a 3D point p_{3d} to a 2D point p_{2d} on the image plane of I.

3.3.3 Bundle-Adjusting Neural Occupancy Field

From now on, we assume we have a set of images with the same geometric shape *and* a rough pose for every image within each cluster. Directly applying a variation

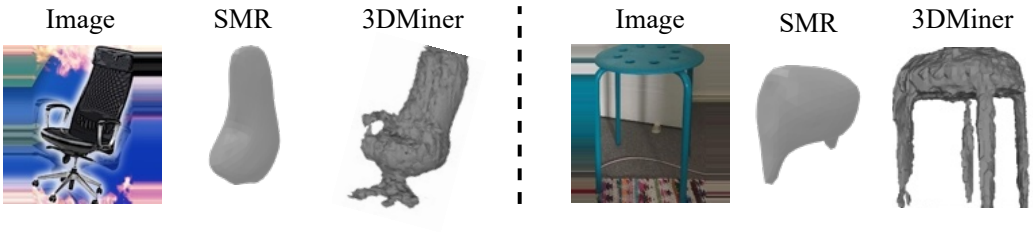

Figure 3.3: **Qualitative Comparison on Pix3D chairs.** When applied to actual in-the-wild images of objects, 3DMiner generates much more accurate geometry than state-of-art methods like SMR.

of NeRF (Mildenhall et al., 2020) is not a viable approach as there exist no photo-consistency due to the difference in textures and backgrounds. Therefore, we propose to use silhouettes instead of RGB images. Current foreground extraction techniques are very mature and generalize well. We use IS-Net (Qin et al., 2022) to perform foreground segmentation in each image. The DINO-ViT features can also be used to do foreground segmentation, but we found that IS-NET provides more accurate segmentation masks.

We draw inspiration from BARF (C.-H. Lin et al., 2021) and optimize an implicit occupancy field with bundle-adjustments. To do so, we first need to convert the estimated orthographic cameras into perspective cameras.

Orthographic to perspective initialisation. For every image, we use the orthographic parameters from M (estimated with Equation 3.2) to initialise a perspective camera-to-world matrix P in $\mathbb{R}^{4\times4}$:

$$P = \begin{bmatrix} \frac{m_1}{||m_1||} & | \\ \frac{m_2-(p_1\cdot m_2)p_1}{||m_2-(p_1\cdot m_2)p_1||} & T \\ p_1 \times p_2 & | \\ 0 & 1 \end{bmatrix}^{-1}, \tag{3.3}$$

where m_i and p_i denote the i^{th} row of M and P, respectively. To understand

this derivation, recall that the top-left sub-matrix $P_{[1:3,1:3]}$ is the rotation matrix controlling the camera viewing direction, while M, the orthographic projection matrix, is a $\mathbb{R}^{2\times3}$ matrix formed by two orthogonal 3D vector and can be interpreted as a linear plane of projection. The rotation corresponding to M is thus simply obtained by applying the Gram-Schimdt orthonormalization process for the first two rows, and getting the cross product for the third, as suggested by (Zhou, Barnes, Lu, Yang, & Li, 2019). Note that M, which we estimate by Rigid Factorization, is initially orthogonal, so we could simply set $P_{[1:2,1:3]}$ to be a normalized version of M. However, we seek to optimize M via gradient descent, which does not guarantee that M remains orthonormal throughout the process, hence why Gram-Schimdt orthonormalization is important. T is a translation vector initialised as $[0, 0, z]^T$, where z is a scalar hyperparameter (set to 5 for our experiments). We also initialise a camera intrinsic matrix K with focal length f equaling to the image size. This initialization, though inaccurate, is optimized during the bundle-adjustment.

Bundle-adjusting camera parameters. Given K and P, we cast rays through each pixel, and sample points along each ray r. Each point x is encoded with the progressive positional encoding technique from BARF (C.-H. Lin et al., 2021) where the k^{th} frequency of the positional encoding is:

$$\gamma(x, \alpha) = w(\alpha) \cdot [\cos 2^k \pi x, \sin 2^k \pi x], \tag{3.4}$$

where w is a weight controlled by hyperparameter α that gradually increases as the training progresses. This effectively activates the encoding of higher frequencies as training progresses.

We feed the positional encoded inputs into the occupancy field MLP and obtain an occupancy output. The loss is a binary cross-entropy comparing the ground-truth silhouettes occupancy o_{gt} and the soft maximum occupancy of the corresponding

ray:

$$\mathcal{L}_r = \text{BCE}(o_{gt}, 1 - e^{(-\sum_{x \in r} \text{MLP}(x, [\gamma(x,\alpha)]_{\alpha=0}^{10}))}). \tag{3.5}$$

We jointly optimize the 3D occupancy network, and the bundle-adjustment parameters f, M, and Ts for each image. We do not directly optimize the matrix P via gradient descent because it needs to remain in the manifold of rotation matrices. We follow (C.-H. Lin et al., 2021) and use marching cube to extract a mesh from the learned occupancy field.

Regularizing the geometry and space For real-world datasets like LAION-5B, the number of images sharing common objects may be very limited. Thus, we draw inspiration from RegNeRF (Niemeyer et al., 2022) and impose two extra regularizations during our occupancy field optimization. First, we encourage piece-wise smoothness of objects by imposing an additional geometric regularizer $\mathcal{L}_g$:

$$\mathcal{L}_g = (d(r_{i,j}) - d(r_{i,j+1}))^2 + (d(r_{i,j}) - d(r_{i+1,j}))^2, \tag{3.6}$$

where $r_{i,j}$ indicates the ray casted from pixel coordinate (i, j) and d is expected depth calculated in the same manner as (Niemeyer et al., 2022). Second, since all our reconstructed shapes can be placed at the center of the coordinate system, we impose space annealing to confine the near and far plane then gradually expanding it as training iteration progresses.

3.4 Experiments

3.4.1 Implementation Details

All our models are trained on a single Tesla V100 machine. We use 10 augmentations for our image clustering step. For keypoints, we use the 8 most up-voted

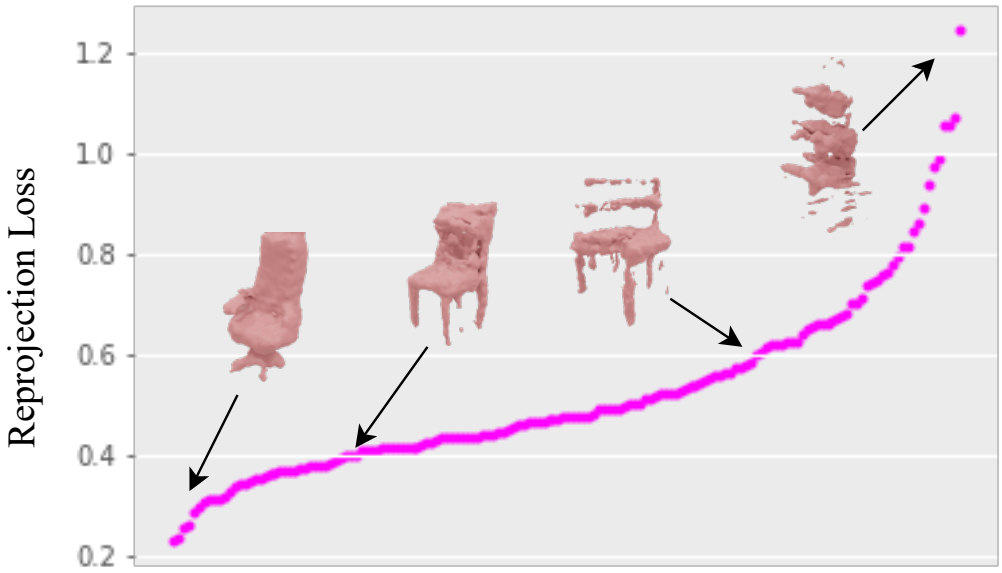

Figure 3.4: **Reprojection error.** We plot the reprojection error per cluster (averaged over each image in the cluster) in ascending order and show representative reconstructions for four data points. We empirically observe that the reprojection error is a good indicator of the quality of the reconstruction.

segments unless specified. We sample 32 rays and 32 points on each ray during every iteration and train the model for 300 epochs with α increasing from 1 every 20 epoch up to 10, using an Adam optimizer with a learning rates of 10^{-3}. The learning rate for camera parameters decay by a factor of 0.1 every 100 epochs.

3.4.2 Comparison on Pix3D chairs

To validate our approach, we directly train the model on the very challenging Pix3D chairs dataset (Sun et al., 2018) consisting in 3839 real images of 561 different chairs. This is the most difficult Pix3D category and has been used by prior work as the benchmark (Duggal & Pathak, 2022; H. Xie et al., 2020) (please see Appendix for additional Pix3D categories). We compare against two state-of-

Figure 3.5: **Qualitative Results on Pix3D and LAION-5B.** We show examples of mining 3D shape from the in-the-wild LAION-5B dataset with the above text prompts. 1∼6 show reconstructions with high fidelity when clusters are very accurate. 7∼10 present generic shapes captured when the clusters are more diverse. 11∼12 are results on very challenging non-rigid objects.

the-art approaches: SMR (T. Hu et al., 2021) and Unicorn (Monnier et al., 2022). Since our method is completely unsupervised and consists of an optimization for every cluster of images, there is no learning involved. Hence, there is no need to split a training and testing dataset. We retrain SMR (T. Hu et al., 2021) and Unicorn (Monnier et al., 2022) on the entire set of images, and evaluate them on the same set. Therefore, all three methods receive the same amount of information during weight optimization.

To evaluate 3DMiner, for each image, we query which cluster it belongs to, and map the image to the corresponding 3D reconstruction. For all methods, we compare the 3D reconstruction in a canonical orientation against their ground truth 3D shape, and average performances across all input images. To focus the evaluation on the quality of the geometries, we align the reconstructions to their ground truth with Coherent Point Drift (Myronenko & Song, 2010), optimizing for translation,

rotation and uniform scaling. SMR requires image masks, while Unicorn only needs an image. To make the comparison fair, we also use masked images for Unicorn and use ground truth masks instead of estimated masks in the last step of our pipeline.

As shown in Table 3.1, our approach greatly outperforms SMR and Unicorn, reducing the Chamfer Distance by 32% and improving the F-score by 10 points. We found that training Unicorn led to a degenerate solution where the network always predict the same mean shape. This aligns with the author's feedback on the official GitHub implementation [1], suggesting that the model is very difficult to train on real-world datasets. By contrast, our meshes are instance-specific and therefore more accurate on a variety of chairs. In Figure 3.3, we provide a short qualitative comparison of 3DMiner against SMR (please see Appendix for more). Additional results for our method in chairs and tables are also presented in Figure 3.5.

Filtering 3D shapes. To mine 3D data from images, it is crucial to automate the process of distinguishing good reconstructions from bad ones. We hypothesize that the reprojection error, which is the final converged loss of our bundle-adjusting reconstruction (see Equation 3.5), correlates with the reconstruction quality. To validate this observation, we show a couple of experiments. First, for each cluster in Pix3D Chairs, we plot the reprojection error averaged per image and show representative rendered meshes in Figure 3.4. An error smaller than 0.4 generally indicates good consistency across all views (22% of the Pix3D clusters). This consistency drops significantly as the error increases, and degenerate solutions generally appear when the error goes over 0.8 (8% of the Pix3D clusters). Second, in the quantitative study on Pix3D in Table 3.1, we select images whose cluster has

[1] https://github.com/monniert/unicorn

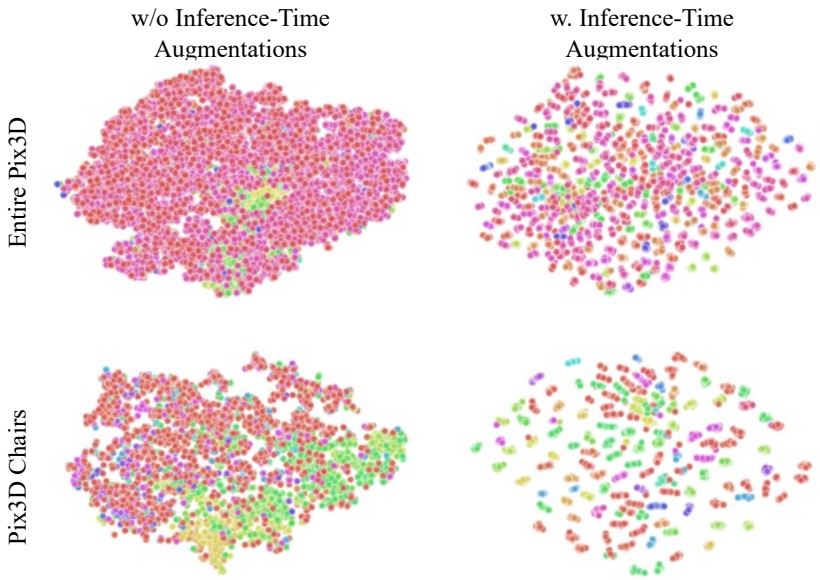

Figure 3.6: **Ablation of our augmentation scheme.** We show the T-SNE embeddings of Pix3D images embedded with DINO-ViT (Caron et al., 2021) with and without our augmentations scheme. Each color denotes a different object. Note that with augmentations, images corresponding to the same 3D object are grouped together, forming clusters on a single color (right), which is not the case without augmentations (left).

a lower reprojection error than a certain threshold, and average the reconstruction error only on this subset of images. The results show that the reconstruction quality improves significantly as we decrease the threshold. As reference, we also present the results of other baselines on the same subset of images.

3.4.3 In-The-Wild Dataset: LAION-5B

To showcase the capability of our 3DMiner, we present, to the best of our knowledge, the first 3D reconstruction results from images in LAION-5B. Lower part of Figure 3.5 shows our results on 12 categories. We download the first 500 images returned from LAION-5B using various text prompts [2], then run 3DMiner on the resulting datasets, with varying distance thresholds (*i.e.,* different numbers of

[2]https://rom1504.github.io/clip-retrieval/

images within the clusters). We filter out the reconstructions with reprojection error > 0.4 per the analysis in the previous section.

As we can see in Figure 3.5, the images from LAION-5B are noisy, but our clustering leads to cleaner subsets of images. Specifically, reconstructions `1`$\sim$`6` have high fidelity as clusters are generally very clean, even when the color and backgrounds vary drastically. Intriguingly, when the clusters are less precise (*e.g.,* reconstructions `7`$\sim$`10`), 3DMiner still captures the generic shape. In particular, `10` clustered images of one and two dumbells, but the bundle-adjustment allows us to find an angle where both images are valid. Finally, we also investigated how the method would perform for non-rigid objects (`11`$\sim$`12`). As expected, our method has a lot of trouble dealing with non-rigid shapes, but ends up reconstructing almost-rigid portions of the geometry (e.g head of the giraffe).

Failure cases on LAION-5B. 3DMiner fails on several text prompts. We identified the two sources of failures: *(i) Not Enough Angles.* Prompts like `fish` exhibit only the side and not the front/back view. This makes it hard to estimate the depth of the object and update the focal length and camera translation accordingly. *(ii) Watermarks* appear in a large portion of LAION-5B data. Our estimated masks accidentally capture the watermarks as salient objects, which severely undermines our reconstruction results. Failure prompts include `Statue of Liberty`.

3.4.4 Analysis

Augmentations before image clustering. We validate the contribution of our augmentation scheme on Pix3D dataset. Figure 3.6 shows a T-SNE visualization of the embedding space created with and without the augmentation. The colors denote the ground truth clusters. Our augmentation scheme clearly helps the DINO-ViT features to be more instance-specific, even when the images come from multiple categories. We further quantify the quality of our clusters by measuring

Method	Dist. Thres.	NMI
Original Image	10	0.647
Image + Aug	10	0.758
Image + Aug	30	0.786
Image + Aug	100	0.764

Table 3.2: **Normalised mutual information comparisons**. We compare the Normalised Mutual Information (NMI) of the ground-truth labels against our predicted clusters. "Aug" denotes our augmentations scheme (see Section 3.3.1), and "Dist. Thres." denotes the threshold distance set for agglomerative clustering. Our augmentation scheme improves the quality of the cluster (+11 points).

Figure 3.7: Two clusters from text prompt `Umbrella`, showing non-rigid objects separated based on part-wise poses.

the normalised mutual information (NMI) with the ground-truth clusters. Table 3.2 shows that adding augmentations improves the NMI by 0.11, and that the quality of the clusters remain stable when we vary the distance threshold used during agglomerative clustering. This is a useful feature when applying 3Dminer to real-world datasets: we can automatically test several distance thresholds to gather enough images within each cluster while maintaining cluster precision.

Clustering Non-rigid Objects. One additional benefit of doing the clustering is that it tends to group objects where the part-wise poses are also similar. We show in Figure 3.7 an example taken from the prompt `Umbrella`, where opened and closed umbrellas are grouped into different clusters, allowing us to reconstruct the object even with certain level of non-rigidity (The right cluster is what allows the reconstruction of umbrella shown in Figure 3.5). Tightening the clustering threshold would also constrain rigidity within a cluster, though with the trade off

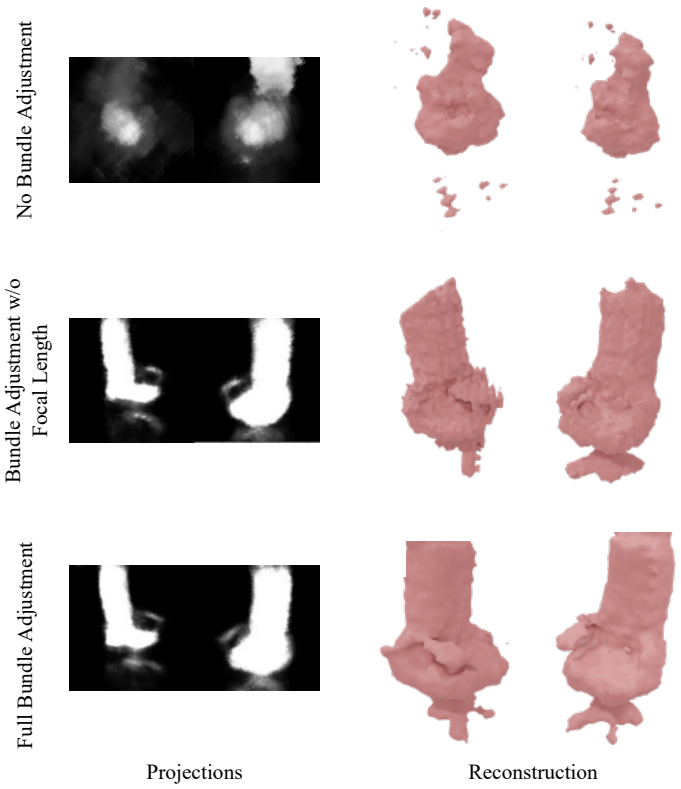

Figure 3.8: **Ablations on Bundle-adjusting poses and focal lengths**. We visualize the results of bundle adjustment on the reprojection error and overall reconstruction quality

of fewer images and thus a higher likeliness of degenerate 3D reconstructions (*e.g.*, most cases of clusters with only 3 or 4 images tend to fail).

Ablations on Bundle-adjustment We hope to understand the capability of our orthographic-to-perspective bundle adjustment. To rule out any uncertainties we select one Pix3D cluster that is 100% accurate (*i.e.,* all images depict the same 3D shape). Our reconstruction achieves roughly 0.38 F-Score. We present the reconstructed meshes under three training settings: *(i)* no bundle-adjustment at all, *(ii)* bundle adjusting only the motion and translation, and *(iii)* bundle adjusting all camera parameters together.

As shown in Figure 3.8, even though the initial rigid factorization algorithm constrains all cameras to point to the same shape, the poses are very rough at the

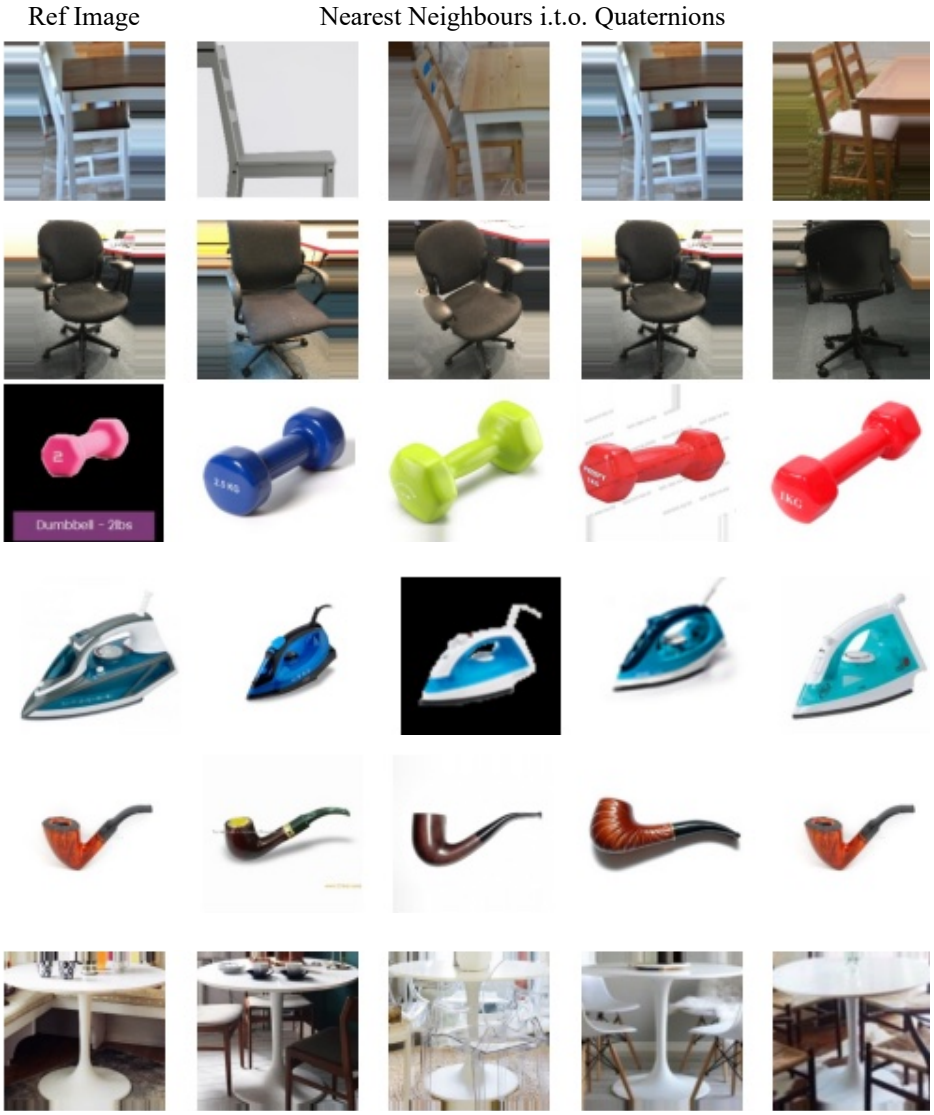

Figure 3.9: **Pose estimation Analysis.** We randomly select a reference image (leftmost column) out of a cluster and find the images with the nearest quaternion poses (rightmost 4 columns).

start. This leads to a very coarse and degenerate solution (without any handles and legs). As we introduce motion and translation, we gain significantly more details of the chair. Nevertheless, due to the focal length inaccuracy, we may end up with artifacts (*e.g.,* only one armrest, missing legs) or even rougher edges due to inconsistencies between the projections. Learning the focal lengths is thus

important to recover details in the geometry.

Pose Estimation on Similar Objects. In addition to shapes, 3DMiner also provides association images and *posed shapes* – shapes that are aligned with the image content when the estimated pose is applied. We show a qualitative evaluation in Figure 3.9. Specifically, given a cluster, we take a random image for reference (leftmost column), and find the 4 nearest neighbours in terms of their quaternion poses within the cluster (right 4 columns). We show that even when the shape, texture, and background differs, our bundle-adjustment still allows us to capture a rough pose among them leading to promising shape reconstructions.

3.5 Conclusion

We have presented 3DMiner, a novel pipeline for mining 3D shapes from large-scale unnanotated in-the-wild image datasets. Differently from single network end-to-end approaches, our technique can be thought of as a reincarnation of classical approaches (Vicente et al., 2014) while replacing manual annotations with representations learned from deep networks. The key elements of our pipeline are: *(i)* a clustering step using DINO-ViT features, *(ii)* a camera estimation step using a classical Structure-from-Motion technique and keypoint estimation, and *(iii)* a progressive bundle adjusting reconstruction to learn an occupancy field supervised by image silhouettes. Through rigorous experiments, we have showed that 3DMiner outperforms the state-of-the-art on the Pix3D dataset, and to the best of our knowledge, is the first to show 3D reconstruction results on the LAION-5B dataset. We hope the 3DMiner serves as a testbed for a newly proposed task of mining geometry from large-scale unannotated datasets and all the subproblems it involves.

Limitations. Although our model is able to reconstruct 3D shapes solely from

in-the-wild images, the concavity of the shapes is not captured. This is because the occupancy field is supervised with a silhouette loss, which amounts to space carving. Future works could explore using monocular depth estimation networks as further supervision in the reconstruction problem. Furthermore, 3DMiner is a sequential pipeline and thus vulnerable to the failure of any step. Fortunately, we verified that we can automatically identify bad 3D reconstructions in order to discover only reasonable 3D shapes. As every component of the pipeline eventually advances, we hope that the number of meaningful 3D shapes discovered from image datasets can be increased. Finally, we also hope to investigate the utilization of using 3DMiner generated shapes for supervision in training better SVR techniques.

Acknowledgements. This work is supported in part by the EPSRC ACE-OPS grant EP/S030832/1.

3.6 Additional Step-by-Step Visualization

In this supplementary material, we provide additional comparisons on Pix3D tables and sofa, as well as qualitative visualizations on every step of our 3DMiner pipeline with a Pix3D and a LAION example.

3.6.1 Clustering Images with Similar Objects

This corresponds to applying the method described in Section 3.1 of the main paper. The main idea of this experiment is to show how the clustering procedure groups images depicting similar shapes together, even when presented with heavy nuisance factors – objects have different colors, materials, textures and in completely different environments. As a reference, we present subsets of the unclustered images in Figure 3.10 As we can see, creating representations through image augmentations followed by feature pooling leads to representations that

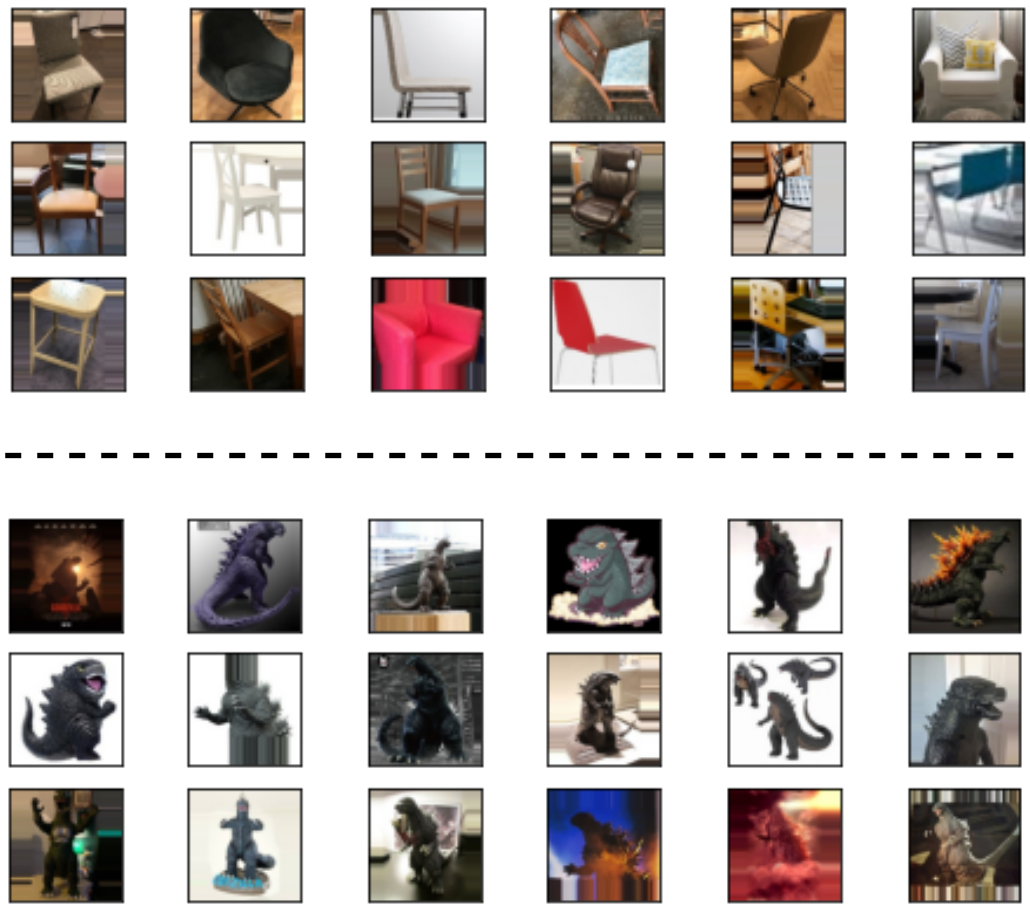

Figure 3.10: **Samples of the Original Image Set**. Image above the dotted line are from Pix3D and below are from LAION-5B Godzilla. You can easily spot the noisiness within the original datasets.

are robust to these nuisance factors and mostly encode shape information. We show three examples each of clusters from the two datasets, Pix3D and LAION-5B `Godzilla`, in Figure 3.11 and Figure 3.12, respectively.

3.6.2 Initial Pose Estimation

This section illustrates the method described in Section 3.2 of the main paper. It is mainly divided in two parts. The first one analyzes local image features from all the images groups their pixels according to feature similarity. This establishes

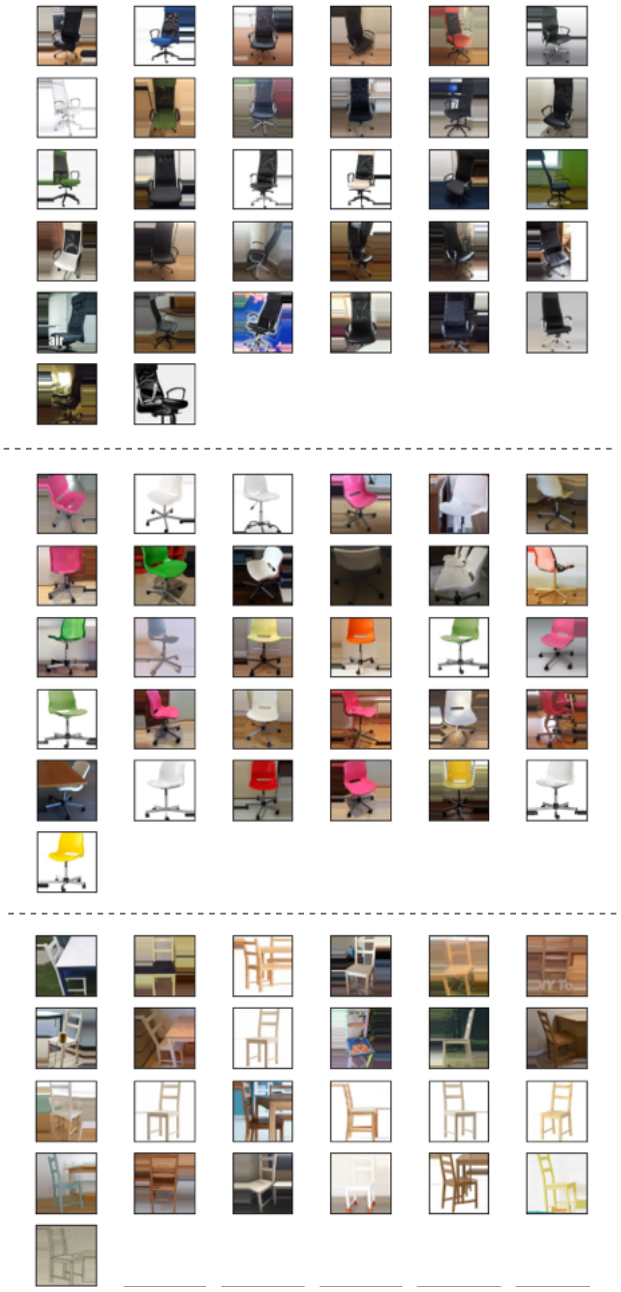

Figure 3.11: **Example Clusters from Pix3D.** Each set of images separated by dotted lines shows an image cluster computed from Pix3D chairs using the method described in the main paper in Section 3.1. Notice how even when the have objects in different colors, textures, completely different backgrounds and illumination, one set of images still roughly depicts objects with the same geometry.

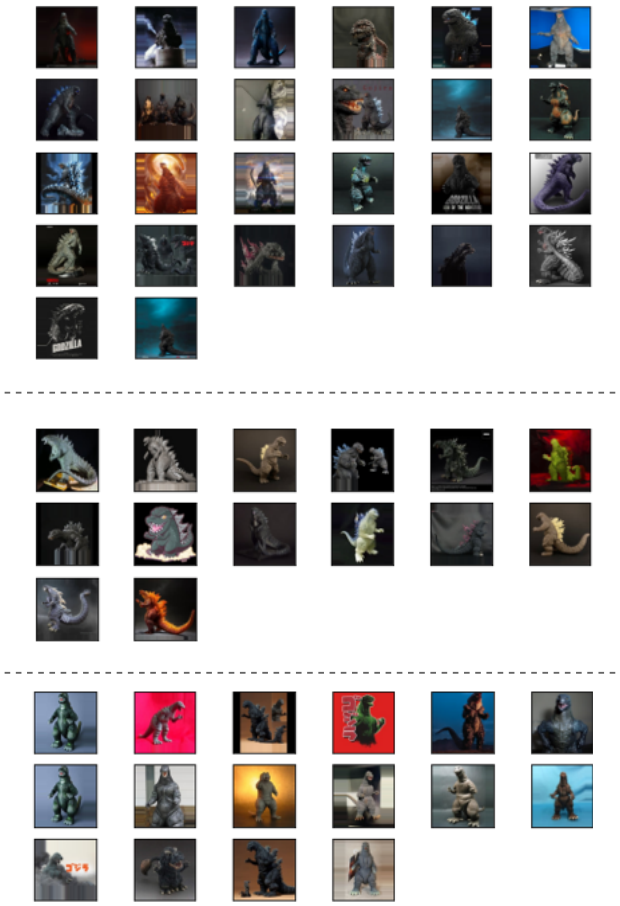

Figure 3.12: **Example Clusters from LAION-5B.** Each set of images separated by dotted lines shows an image cluster computed from LAION-5B using the prompt `Godzilla`. Clusters were computed using the method described in the main paper in Section 3.1.

correspondences between pixels of different images. The second step uses the aforementioned correspondences in a rigid factorization procedure and yields an orthographic camera pose for every image within the cluster.

3.6.2.1 Part Segmentation and Masks

We present part segmentations for both Pix3D and LAION-5B in Figure 3.13. **The center point of the part segmentations are the keypoints**. The part segmentation is performed by extracting the dense features from DINO-ViT then

Figure 3.13: **Part Segmentations.** We visualize the part segmentations of a Pix3D chair and a LAION-5B Godzilla cluster. Note that the Godzilla images goes through ISNet for background removal (First two columns) before the part segmentation. We realized that the Pix3D part segmentations work better with background. Notice that the part segmentations are far from perfect but their overall semantics is consistent across all the images in the cluster. For example, all the office chairs have the wheels labeled with green parts, whereas most of the godzillas have their legs labeled with purple parts, brown heads, and so on.

finding common features across images by K-means clustering. Note that the first two columns depicting the LAION cluster have already gone through ISNet for background removal (for Pix3D we use ground truth masks). We observed that keeping the background yielding visually more appealing part segmentations for

Pix3D Chair LAION-5B Godzilla

Figure 3.14: **Orthographic Pose Initialization.** We select 3 anchor image (left column) and output the set of images that are in nearest quarternion distance with the anchors.

Pix3D especially under the cases of occlusion.

3.6.2.2 Orthographic Pose Estimation

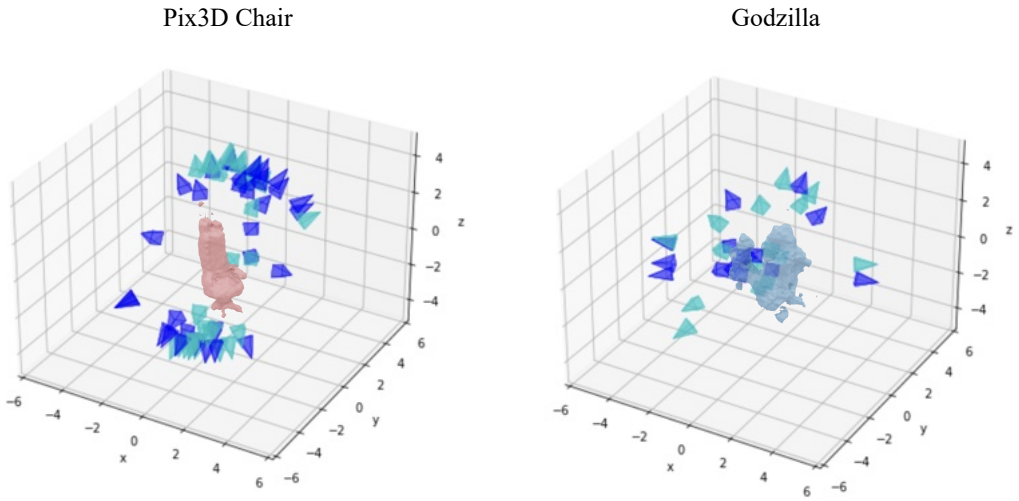

Figure 3.15: **Camera Poses after Bundle Adjustment**. We visualize the cameras before (blue) and after bundle adjustment (cyan).

We use the previously computed correspondences to estimate orthographic camera poses using rigid factorization to obtain a reasonable initial pose, illustrated in Figure 3.14. For this image, find three random images per cluster and present the 4 nearest neighbors in terms of their quaternion distance. We can see that, even

though the estimations are far from perfect, the poses serve as good initial points to the orientation of the object.

3.6.3 Shape Estimation & Bundle Adjustment

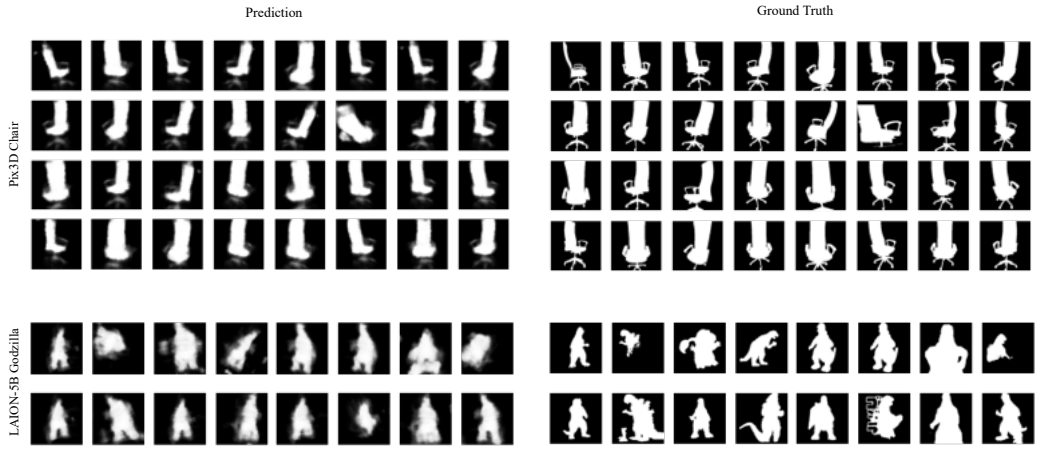

Figure 3.16: **Reprojections.** We visualize the reprojections against the ground truth for the Pix3D chair and LAION-5B Godzilla.

Finally, we show the quality of our occupancy field through Figure 3.16 and 3.15. Figure 3.16 shows the reprojection errors after training, and Figure 3.15 depicts the changes in the camera poses through bundle adjustment.

4 | Learning Continuous 3D Words for Text-to-Image Generation

The main content of this chapter is published and presented in CVPR 2024.

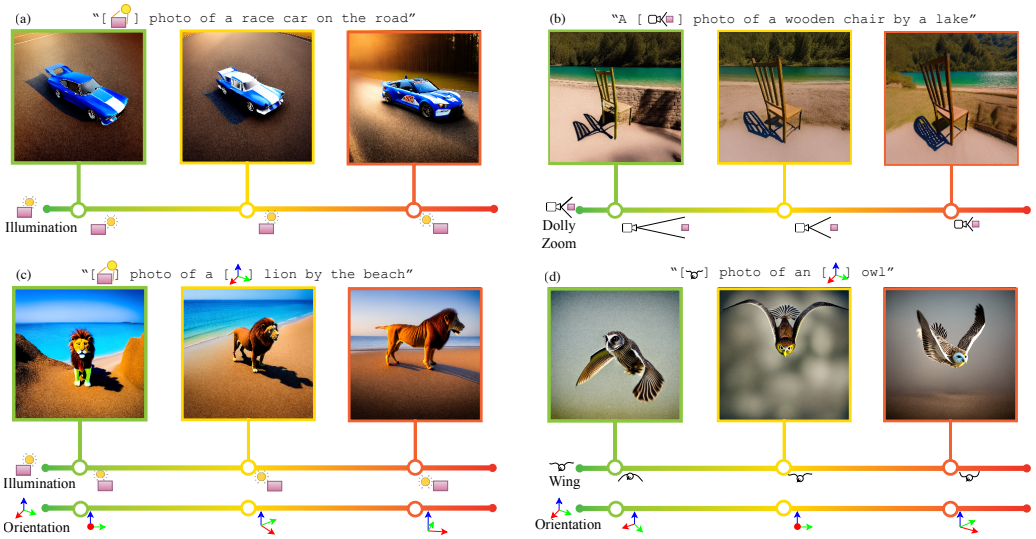

Figure 4.1: We introduce **Continuous 3D Words** – special tokens in text-to-image models that allow users to have fine-grained control over several attributes like illumination [ ] (a and c), non-rigid shape change [ ] (d), orientation [ ] (c and d), and camera parameters [ ] (b). Our approach can be trained using a *single 3D mesh* and a rendering engine while incurring negligible runtime and memory costs.

In this chapter, we explore the possibilities of learning and controlling 3D-aware concepts beyond geometry from large-dataset-pretrained models. Current controls over diffusion models (e.g., through text or ControlNet) for image generation fall short in recognizing abstract, continuous attributes like illumination direction or non-rigid shape change. In this paper, we present an approach for allowing users of text-to-image models to have fine-grained control of several attributes in an image. We do this by engineering special sets of input tokens that can be transformed in a continuous manner – we call them **Continuous 3D Words**. These attributes can, for example, be represented as sliders and applied jointly

with text prompts for fine-grained control over image generation. Given only a single mesh and a rendering engine, we show that our approach can be adopted to provide continuous user control over several 3D-aware attributes, including time-of-day illumination, bird wing orientation, dollyzoom effect, and object poses. Our method is capable of conditioning image creation with multiple Continuous 3D Words and text descriptions simultaneously while adding no overhead to the generative process. Project Page: https://ttchengab.github.io/continuous_3d_words

4.1 Introduction

Photography is fascinating because it enables very detailed control over the composition and aesthetics of the final image. On the one hand, this is simply the result of application of physical laws to achieve image acquisition. On the other, the slightest changes in the moment captured, illumination, object orientation, or camera parameters bring a completely different feeling to the viewer. While the giant leap of modern text-to-image diffusion can bring generated 2-D images to close proximity with real photos, text prompts are inherently limited to high-level descriptions, far removed from the detailed controls one has over actual photography. This is mainly due to the scarcity of such descriptions in the training dataset — very few would describe a photo based on exact object movements and camera parameters like the wing pose of a bird or the rotation of a person's head in degrees. On the other hand, 3D rendering engines allow us to mimic many of these 3D controls that photographers enjoy. We can render images of objects with predefined camera, illumination and pose changes at a very fine-grained scale. However, creating detailed 3D worlds is incredibly laborious, which limits the diversity of the scenes that can be generated by non-specialized practitioners. In that regard, using text-to-image diffusion to create images is a much more accessible technology,

whereas precise 3D scene control remains firmly in the domain of experts.

In this work, we aim to bring together the best of two worlds by expanding the vocabulary of text-to-image diffusion models with very few samples generated from rendering engines. Specifically, we render meshes based on the attribute we aim to control, creating images with color and other useful information to generate a small set of data samples. The goal is to disentangle these abstract attributes from the original object and encode them into the textual space in a controllable manner – we term these attributes *Continuous 3D Words*. They allow users to create custom sliders that enable fine-grained control during image generation and can be seamlessly used along text prompts.

At the heart of our approach is an algorithm to learn a continuous vocabulary. The benefits of continuity are two-fold: *i)* the association between different values of the same attribute makes it much easier to learn, rather than having to learn hundreds of discrete tokens as an approximation and *ii)* we learn an MLP that would allow interpolation during inference to generate an actual continuous control. On top of this, we also propose two training strategies to prevent degenerate solutions and enable generalization on new objects beyond the training category. First, we apply a two-stage training strategy: we first apply the Dreambooth (Ruiz et al., 2023) approach to learn the object identity of the underlying mesh used for rendering, then sequentially learn the various attribute values disentangled from the object identity. This prevents the model from falling into a degenerate solution of encoding each value of an attribute as a new object, which would prevent us from generalizing the attribute to new objects. Second, we apply ControlNet (L. Zhang et al., 2023) with various conditioned images to generate a set of additional images with varying backgrounds and object textures. This prevents the model from overfitting to the artificial backgrounds of rendered images. The entire training was done in a light-weight Lower Rank Adaptation (LoRA) (E. J. Hu et al., 2021) manner,

making it fast and accessible with single GPUs.

We implement our continuous vocabulary and training method, across various sets of single (e.g., dollyzoom extracted from chairs) and multiple (e.g., object pose and illumination extracted from a dog mesh) attributes, and show through quantitative user studies and qualitative comparisons that our method can properly reflect various attributes while maintaining the aesthetics of the image — significantly outperforming competitive baselines.

In summary, we present 1) Continuous 3D Words, a new method of gaining 3D-aware, continuous attribute controls over text-to-image generation that can be easily tailored to a plethora of new conditions, 2) a series of training strategies to disentangle the attributes from object identity to enhance the improvements in image generation and 3) extensive qualitative and quantitative studies to showcase our approach in various interesting applications.

4.2 Related Work

Conditional Diffusion Models. Ever since diffusion models (Ho et al., 2020; J. Song, Meng, & Ermon, 2020) pushed the quality and generalizability of image generation beyond GANs (Goodfellow et al., 2020), the vision community has introduced a diverse range of modalities that can be used as conditions to control the image generation process. The most common condition is currently text. Works such as DALLE (Ramesh et al., 2022, 2021; Betker et al., 2023) and Imagen (Saharia et al., 2022) used large scale text-image datasets and strong language understandings from pretrained LLMs (Brown et al., 2020; Raffel et al., 2020; Devlin et al., 2018) to guide the generation process. Stable Diffusion further popularized this class of methods by employing memory efficient models through latent-space diffusion (Rombach et al., 2021). Other works built on top of these

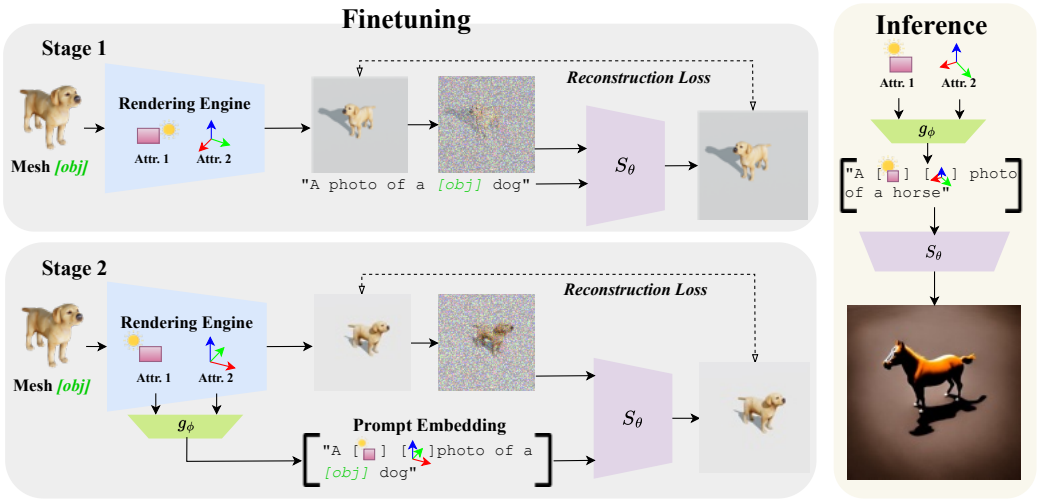

Figure 4.2: **Method Overview.** *Finetuning:* Our finetuning is divided into two stages. In the first stage, we render a series of images using different attribute values (e.g., illumination and pose). We feed them into the text-to-image diffusion model to learn token embedding *[Obj]* representing the *single mesh* used for training. In the second stage, we add the tokens representing individual attributes into the prompt embedding. The two stage training allows us to better disentangle the individual attributes against *[Obj]*. *Inference:* Attributes can be applied to different objects for text-to-image generation.

models by adding other forms of conditioning (L. Zhang et al., 2023; Mou et al., 2023). Highly relevant to our work is ControlNet (L. Zhang et al., 2023), which proposes a general pipeline with zero-convolutions for conditioning on text and image data (e.g., depth maps, canny maps, sketches). Despite their impressive image quality, it is not clear how to use these models to control other attributes of images like illumination or object orientation.

Other set of works explored how to perform image edits using textual instructions. Given a text-generated image, they demonstrate how the user can edit the image by amending the prompt, yet still preserve some aspects of the original image (Parmar et al., 2023; Hertz et al., 2022; Brooks et al., 2023). While convenient, these approaches do not allow *fine-grained* control over image elements since they are ultimately restricted by the user's ability to describe visual content through text – e.g., it would be very difficult to change the illumination direction by a precise

angle such as 11°.

Recently, as the amount of 3D data available significantly increased, Liu et al. (R. Liu et al., 2023) introduced Zero-1-to-3, a diffusion model trained on various viewpoints of 3D rendering that enables viewpoint editing given an image of a single object. Similarly, works like DreamSparse (Yoo et al., 2023) also employ diffusion models to synthesize novel views on open-set categories. Differently from our approach, these techniques are focused only on object orientation and rely on vast 3D shape datasets. On the other hand, we investigate how to learn several continuous concepts (e.g., illumination [] , wing pose [] , dolly zoom []) that can be directly used in text-to-image scenarios; i.e., we don't generate an image to then change the orientation or illumination later, but instead we use *Continuous Words* directly on text prompts.

Learning new concepts on diffusion models. With diffusion models being trained on unforeseen quantities in images and texts, a stream of work focused on adding specific concepts with very few data samples. For example, given a small set of images representing one particular object instance, textual inversion learns a new word embedding to describe the object, such that the word can be applied with new text prompts for image generation (Gal et al., 2022). NETI (Alaluf et al., 2023) extended the word embedding to a time-space conditioned neural mapper for better generation while preserving quality. Similarly, Dreambooth (Ruiz et al., 2023) aims to achieve the same goal, but by using a repurposed token rarely used in text and finetuning the entire diffusion model with an additional constraint to prevent generative loss. There are numerous subsequent works showing improvements on finetuning different layers/weights and by improving the training strategy (Kumari et al., 2023; Han et al., 2023; Z. Liu et al., 2023).

Despite the advances in adding new personalized entities to existing models, few

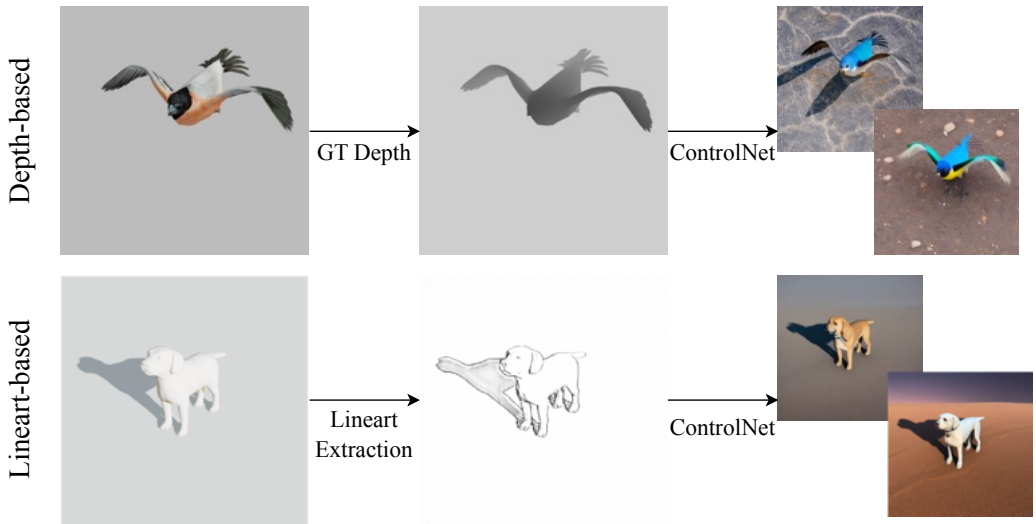

Figure 4.3: **ControlNet Augmentations.** Depth ControlNet is used for attributes creating direct shape changes. Lineart ControlNet is applied for more subtle changes that cannot be reflected by depths (e.g., illumination).

works focus on learning general concepts that can be applied to a variety of scenarios. A concurrent work, ViewNETI (Burgess et al., 2023), is the first to learn viewpoints as a concept, but we hypothesise that the 3D awareness of large text-to-image diffusion models goes far beyond merely viewpoints, allowing us to associate and even create interactions with multiple 3D-aware concepts like illumination, pose and camera parameters *at the same time*. Our method, despite being trained only using a single mesh, shows superior generalization properties – while trained to learn illumination and orientation from renderings of a single dog, we are capable of employing the learned concepts to generate cars, horses (Figure 4.4), polar bears (Figure 4.9), lions (Figure 4.1) and so on.

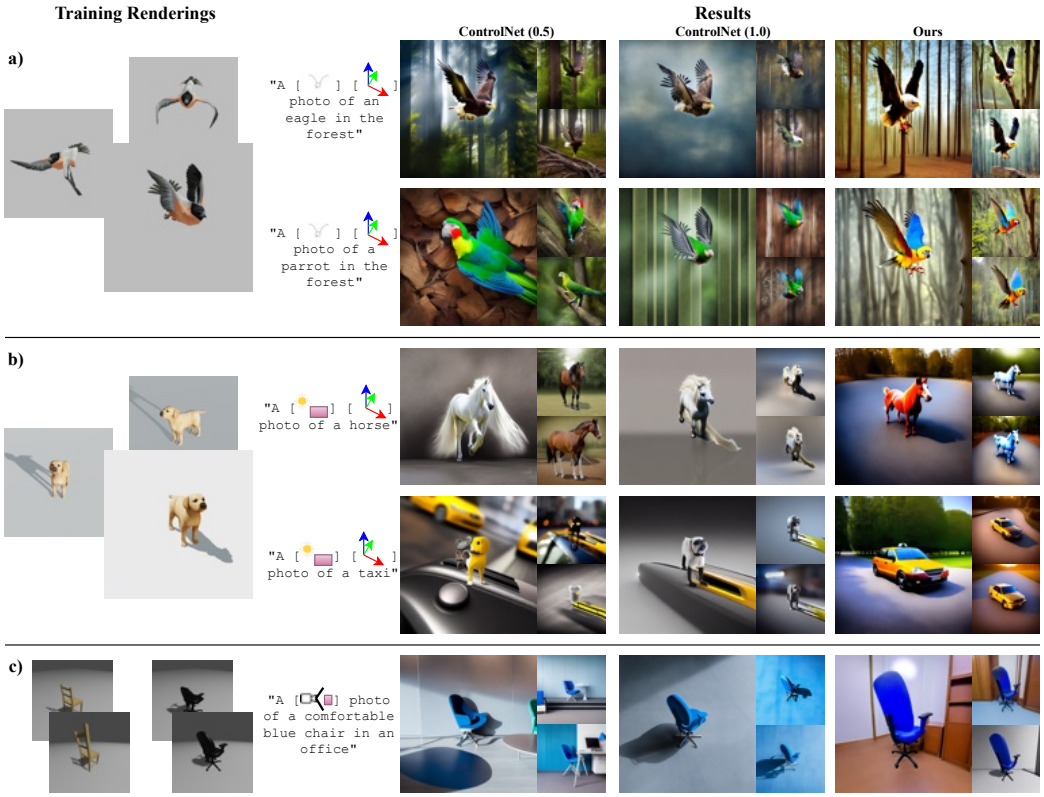

Figure 4.4: **Qualitative Comparisons.** We compare our Continuous 3D Words trained under three settings against ControlNet of various strengths. Note that the dollyzoom setup was trained with trained with multiple chair meshes, so we give additional ControlNet by manually picking the chair rendering that best follows the prompt (i.e., "comfortable" and in "the office").

4.3 Method

4.3.1 Preliminaries

We define an image I that captures object O from category C as a function of several attributes $I = f(a_1, a_2, a_3, ..., a_n)$, where a_i belongs to a vast set of image attributes $\mathcal{A}$: shape, material reflectivity, rotation/translation, camera intrinsic/extrinsics, shape deformations, etc. Some of these components can be translated to other categories while others cannot, so for simplicity we assume they only

work for a single category. In the experimental section of this study, we will demonstrate that the definition of category for some attributes is rather loose and the user is capable of generating images with *continuous words* depicting objects very different from the ones seen during training. Notice that images annotated with the attributes we are interested are very rare, so the models do not have a very precise knowledge of them, except for what is already described in text-image pairs.

Given an image set I^O with images capturing an object O, previous methods like Dreambooth (Ruiz et al., 2023), Custom Diffusion (Kumari et al., 2023), or Textual-Inversion (Gal et al., 2022) aim to minimize the following objective:

$$\mathbb{E}_{\hat{I}_{\epsilon,i},T_O,I_i\in I^O}\left[\left\|S_\theta(\hat{I}_{\epsilon,i},P(T_O))-I_i\right\|_2^2\right], \tag{4.1}$$

where S_θ is a Text-to-Image diffusion model (Rombach et al., 2021), $\hat{I}_{\epsilon,i}$ is a noised image $\alpha_t I_i + \sigma_t \epsilon$ with noise ϵ and noise schedulers α_t, σ_t. $P(T_O)$ is the prompt condition which contains a token embedding T_O used as an identifier of object O. In practice, $P(\cdot)$ is the text encoder from CLIP. The fine-tuned network S_θ can then generate new images containing O when given a new prompt condition $P'(T_O)$ and some Gaussian noise. Unlike previous methods, our goal is not to add concepts representing specific objects, but rather have them describe some attributes $a_i \in \mathcal{A}$, by learning to disentangle them using as few objects as possible within C – most of the time just one object suffices. Our model can also be easily extended to allow control of multiple attributes *at the same time*.

4.3.2 Continuous Control

A naive way to control an attribute a is to use some realistic rendering engine to generate images of the available objects that have the same value $a = x$, and then

apply similar approaches to previous works by assigning a token T_x to identify images with this particular value. This is not ideal, however, since a is often continuous and have infinitely many values – we would require an unfeasible number of tokens to gain fine-grained control over these attributes.

Therefore, we propose to instead learn a continuous function $g_\phi(\mathbf{a}) : \mathcal{D} \rightarrow \mathcal{T}$ that maps a set of attributes from some continuous domain $\mathcal{D}$ to the token embedding domain $\mathcal{T}$. We use positional encoding to first cast each attribute $a \in \mathbf{a}$ to a higher frequency space before feeding into the function, which is represented by a very simple 2-layer MLP. The output of this network is named *Continuous 3D Word* and will allow users to easily control continuous attributes from text prompts augmented by these tokens. Finally, our training objective can then be formulated as:

$$\arg\min_{\theta,\phi} \mathbb{E}_{\hat{I}_{\epsilon,\mathbf{a}},\mathbf{a}}\left[\left\|S_\theta\big(\hat{I}_{\epsilon,\mathbf{a}}, P(g_\phi(\mathbf{a}))\big) - I_\mathbf{a}\right\|_2^2\right]. \tag{4.2}$$

4.3.3 Disentangling Object Identity and Attributes

In practice, when we only utilize a single object to learn attributes, a degenerate solution occurs when directly optimizing (4.2), where S_θ treats the same object with different values for a as different objects. This hinders the generalization capability of changing the attribute when given an image of a new object. To this end, we propose two strategies, one during training and one during inference, to disentangle the object identity and individual attributes.

Training: Learning identity and attributes. We provide a simple regularization by explicitly forcing the model to use the same identifier for images representing the same object. The objective can thus be formulated as:

$$\arg\min_{\theta,\phi} \mathbb{E}_{\hat{I}_{\epsilon,\mathbf{a}},\mathbf{a}}\left[\left\|S_\theta\big(\hat{I}_{\epsilon,\mathbf{a}}, P(T_O, g_\phi(\mathbf{a}))\big) - I_\mathbf{a}\right\|_2^2\right]. \tag{4.3}$$

Optimizing both T_O and $g_\phi(\mathbf{a})$ concurrently proved to be difficult from our experiments (see Section 4.5). We propose a two-stage training strategy as depicted in Figure 4.2. For every available image in I^O with varying $\mathbf{a}$, we first use the same prompt condition $P(T_O)$ to associate them to the same object, then learn the diffusion model parameters θ and our Continuous 3D Words MLP g_ϕ by using the prompt condition $P(T_O, g_\phi(\mathbf{a}))$. The specific rare token for T_O is denoted as *[Obj]*.

Inference: Negatively prompting object identifier. To further the disentanglement of attributes against object identities. We propose a simple trick during inference by adding *[Obj]* as a negative prompt. Specifically, for each sampling step, we swap the null-text embedding used by classifier-free guidance with T_O. Intuitively, since our training happened mostly using O, we want to disincentivize the model to generate images containing such object.

4.3.4 ControlNet Augmentation

To prevent the fine-tuning process from overfitting to a simple white backgrounds and pre-defined object textures, we augment the backgrounds and textures in the rendering process. However, directly doing this in simulation engines is time consuming specially if one is targetting realistic scenes. Thus, we propose an automated solution by utilizing pretrained ControlNets.

Figure 4.3 shows two types of ControlNet augmentations we used in our framework. For attributes that can be directly reflected on shape changes (e.g., wing pose), we directly render the ground-truth depth maps to use as the condition for ControlNet. On the other hand, for attributes that cannot be reflected directly from depths (e.g., illumination), we first render the images without textures, then use a lineart extractor to obtain a “sketch" of the image. This captures subtle changes such as shades and shadows in the pixel space, which can then be used as the condition for

Lineart ControlNet.

We add additional prompts describing the object appearance and background during the ControlNet generation. Both our depth and lineart ControlNet augmentations for training had to be carefully designed as too large of a deviation may lead to wrongly synthesized images. We describe the prompts used for the ControlNet augmentations in our settings 1) wing pose, 2) dollyzoom, and 3) illumination.

Wing pose [ ] . Since the correct position of the wings is particularly important for this case, we engineered the prompts that helped the most in reflecting both wings during generation, which are `{with two wings, flying}`. We also added two time-of-day prompts to increase the variety of backgrounds. Therefore, the overall prompts are generated randomly with: `a bird {with two wings, flying} on a {rainy, sunny} day`.

Dollyzoom [ ] . As there already comprises 5 types of chair inside the training dataset for dollyzoom we focus on augmenting the background with ControlNet. Our overall prompts are generated by: `a chair {in the Acropolis, in a forest, under the snow, on a beach, in Times Square, in a department store}`.

Illumination [ ] . We realize that Lineart ControlNet for shadow generation works best with ControlNet guidance 0.6. However, this weaker strength limits the ability for ControlNet to keep the shadow/illumination consistency if additional backgrounds are described (e.g., adding background descriptions like `in a forest` during Augmentation). Hence, our ControlNet augmentation only focuses on a variety of different dogs. Our overall prompts are generated by: `a {white, gray, brown} dog`.

Multi-Concept Control. When training multi-concept controls for by adding orientation to wing pose/illumination, we still use the same prompts as described above.

We include the ControlNet generated images as a small set of data augmentation, and we use the same prompt which we use to guide ControlNet generation to guide our Stage 2 fine-tuning.

4.4 Experiments

We use off-the-shelf Stable Diffusion v2.1 (Rombach et al., 2021) as the backbone of our method. We resort to the recent Low-Rank Adaptation (LoRA) (E. J. Hu et al., 2021) for the fine-tuning of the denoising U-Net and text encoder, allowing us to train on a single A10 GPU occupying roughly 16GB of memory. Thanks to the low-rank optimization, our models have a very small size (approximately 6MB). Training time varies by the complexity of the single/multiple attributes to learn, but falls within 15k to 20k steps, which generally takes around 3-4 hours in a single GPU. For ControlNet augmentation, we use the official implementation of ControlNet v1.1 (L. Zhang et al., 2023).

We implement our Continuous 3D words under five different attribute settings. For single attributes we implement 1) illumination [] using a single dog mesh, 2) wing pose [] using a single animated dove mesh, and 3) Dolly zoom [] with five Pix3D chairs (Sun et al., 2018). For multi-concepts, we train 4) illumination and object orientation [] + [ ] using a single dog mesh and 5) wing pose and orientation [] + [] using a single animated dove mesh. Settings 1) and 4) use lineart images (Chan, Durand, & Isola, 2022) while the others use depth map to compute the ControlNet background augmentation (see Section 4.3.4).

4.4.1 Comparison with Baselines

Baseline Design. We design a very competive baseline that enables fine-grained attribute control in image generation by combining the mesh training data we used

in our experiments, a rendering engine and ControlNet (L. Zhang et al., 2023). Specifically, we take a *novel* text prompt for a *training* object, and grab a corresponding condition map in the training set rendered with the intended attributes. For example, if the prompt is `a [ 𝆕 ] eagle flying in a forest`, we select the frame of the dove mesh that corresponds to the user-prescribed wing pose [𝆕], render its depth map using a rendering engine and pass it through ControlNet with the same prompt but removing the *continuous 3D word*. The strength of the ControlNet guidance is a critical hyperparameter — increase in strength could increase the accuracy of reflecting the attribute but decrease the robustness to generalize to the text-prompt intended object. Therefore, we present the ControlNet baseline with both full and half strength in terms of guidance. We also explored an interpolation of null-text embedding (Mokady, Hertz, Aberman, Pritch, & Cohen-Or, 2022) baseline, but the results failed to capture even the simplest attributes, so they were omitted from our analysis.

Quantitative Results. Due to the complexity and abstract nature of the attributes we are analyzing, automatically measuring whether a generated image reflects a set of values is a considerable challenge. Thus, following previous papers (Ruiz et al., 2023; Hertz et al., 2022; L. Zhang et al., 2023), we create a user study to evaluate the quality of the generated images while controlling several attributes. For each setting 1, 2, 4, and 5, we randomly sample a set of 3D conditions and prompts, generating over 60 questions per setting. Similarly to the user studies above, we then invite 20 participants, showing them all the given constraints we want the image to pertain (prompts and attributes), and ask them to rank each image from best to worst.

Our prompts used for user study comparison are designed such that objects from very close to very far proximity from the training mesh are tested. For illumination [], our comparisons involve `beagles` (high proximity to the dog

	User Preference (%) ↑				
Method	[wing]	[illum]	[wing] / [orient]	[illum] / [orient]	**Avg.**
ControlNet (1.0)	28.3%	16.2%	35.0%	15.0%	23.6%
ControlNet (0.5)	10.0%	28.8%	12.5%	32.5%	21.0%
Ours	61.7%	55.0%	52.5%	52.5%	55.4%
	Average User Ranking ↑				
Method	[wing]	[illum]	[wing] / [orient]	[illum] / [orient]	**Avg.**
ControlNet (1.0)	2.07 ± 0.70	1.49 ± 0.76	2.20 ± 0.68	1.60 ± 0.74	1.84 ± 0.72
ControlNet (0.5)	1.38 ± 0.66	2.11 ± 0.67	1.38 ± 0.70	2.06 ± 0.76	1.73 ± 0.70
Ours	2.55 ± 0.62	2.40 ± 0.73	2.43 ± 0.67	2.33 ± 0.77	2.43 ± 0.70

Table 4.1: **User study results.** We asked users to rank three images according to their preference and how well they followed the given conditions – text prompt and one or two continuous controls. The controls were described by representative images (i.e., arrows for orientation, shaded sphere for illumination, etc). *See supplemental material for more details*. We evaluate four different control types: wing pose [wing] , illumination [illum] , wing pose [wing] + orientation [orient] , and illumination [illum] + orientation [orient] . Cells colored as red and yellow represent the best and second best method, respectively. Our method was selected as the favorite for the majority of users in all evaluated setups.

mesh),`horse` (medium proximity to the dog mesh), and `taxi, rockets` (low proximity to the dog mesh). Similarly, for wing pose [wing] , our comparisons involve `bird with black head` (similar color to the training bird mesh) to `eagle/parrot in the forest` (different identity with additional background descriptions).

We provide a user study question example in Figure 4.5 on the wing pose and bird orientation.

Table 4.1 shows the results in both the percentage of user preference (image chose as the best one) and the overall ranking. Best results are highlighted as red. Our *Continuous 3D Words* won the majority of votes (over 50%) in all analyzed scenarios. Interestingly, the second best (as highlighted in yellow) alternates between the two guidance strengths of ControlNet. For wing-pose based controls, having a weaker control diminishes the strong prior offered by the depth map,

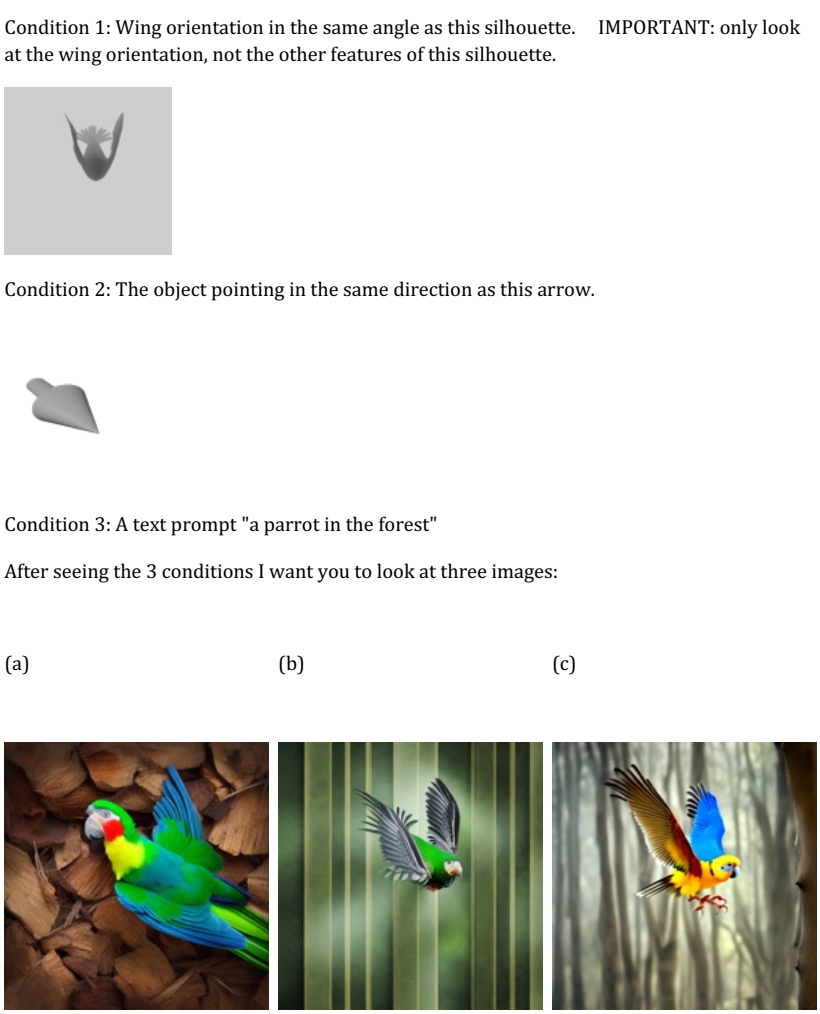

Figure 4.5: Orientation/Wing Pose Survey.

resulting in inaccurate poses. Conversely, strong guidance for illumination forces ControlNet to generate objects around the shadow outline (further shown in Figure 4.4). Differently from the baselines, our training strategy is one-size-fits-all without any hyperparameter required.

Qualitative Analysis We present a detailed comparison of Continuous 3D Words against ControlNet of different guidance strengths in Figure 4.4. We show the results under three training settings: a) wing pose and orientation, b) illumination

and orientation, and c) dollyzoom. We observe that the dollyzoom setup was a harder concept to train and had to be done using five chairs. We also manually "helped" the ControlNet baseline by manually picking the chair that best followed our text prompt as the condition image. More importantly, the ControlNet baselines significantly deteriorate when the prompt contains elements that were not present in the training data. For example, even when trained on a single dog mesh, our method can learn illumination and orientation attributes that can be used to generate horses and taxis (see row *b* in Figure 4.4).

4.4.2 Multi-Concept Control

Just like sentences where we can encompass multiple words, but each disentangled from one another when controlling the image generation, our Continuous 3D Words can do the same. We show four examples in Figure 4.6, two from [] and [] , and two [] and [] , where we can keep one attribute fixed but change the other without sabotaging the quality of image generations. They can be jointly used with complex prompts describing the background and object texture. Moreover, while all these words are learned from a single mesh, we can easily transfer the attribute to objects with fairly close semantics (e.g., a labrador mesh to a polar bear, or a dove mesh to a parrot).

4.4.3 Real World Image Editing

Our Continuous 3D Words can be directly applied onto real world images to perform editing. To do so, we simply have to encode a real world image to a rare token via Dreambooth (Ruiz et al., 2023). Then, we only have to use that token in conjunction with our Continuous 3D Words to generate the edited image. We show 8 examples in Figure 4.7, changing [] , [ ] , [ ] , where we can

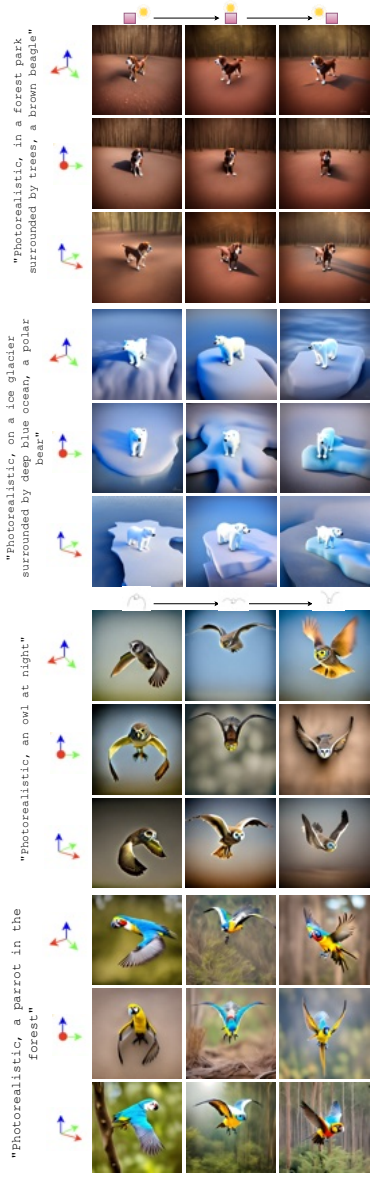

Figure 4.6: **Disentangling Multiple Attributes.** We show four examples of controlling multiple Continuous 3D words in addition to text descriptions. The first 6 rows were trained with a single golden retriever mesh, while the bottom 6 were trained with a single animated dove.

see that the Dreambooth token preserves most structures of the image, while our Continuous Words understands and brings edits to the main subject.

Comparing with Zero-1-to-3. While our approach is focused on enhancing text-

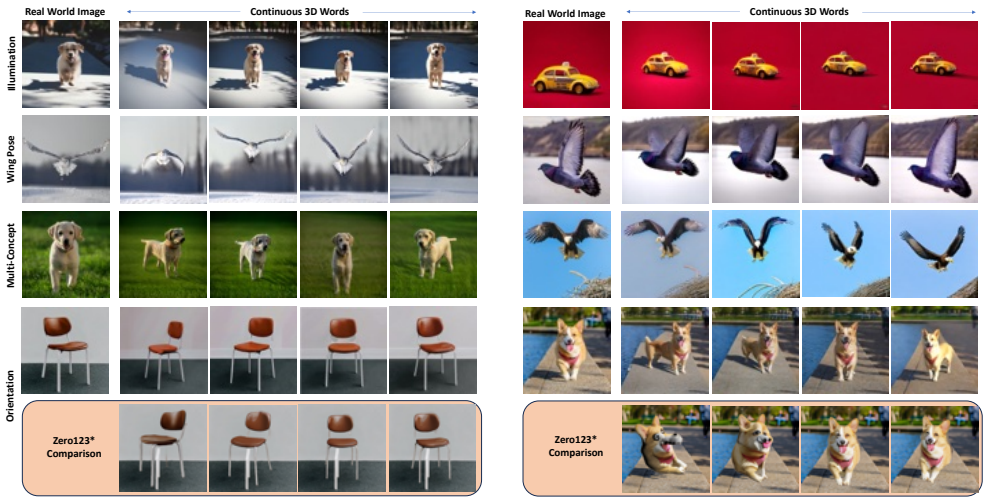

Figure 4.7: **Real World results & Comparison w/ Zero1-to-3.** We learn tokens representing single images to use along with *Continuous 3D Words* for image editing.

to-image, given the capabilities of real-world image editing, we can use the same setup to provide a comparison *only* with object orientation changes (Results in Figure 4.7). Notice that Zero1-to-3 (R. Liu et al., 2023) operates in foreground-only images so we also had to segment the object, inpaint the background and place the novel orientation back into the image. Each one of these steps contain errors that, when compounded, hurt the quality of the final result. More importantly, our method allows controls beyond orientation changes without relying on massive 3D datasets.

4.5 Discussion and Limitations

Why Not Discrete Tokens? The benefits of fitting a single MLP instead of multiple tokens on an attribute with different values are two-fold. First, the MLP learns a continuous function, allowing us to interpolate between two training data samples whereas fitting multiple tokens leads to two datapoints as two discrete mappings. Second, finetuning a model to learn multiple custom tokens simultaneously is

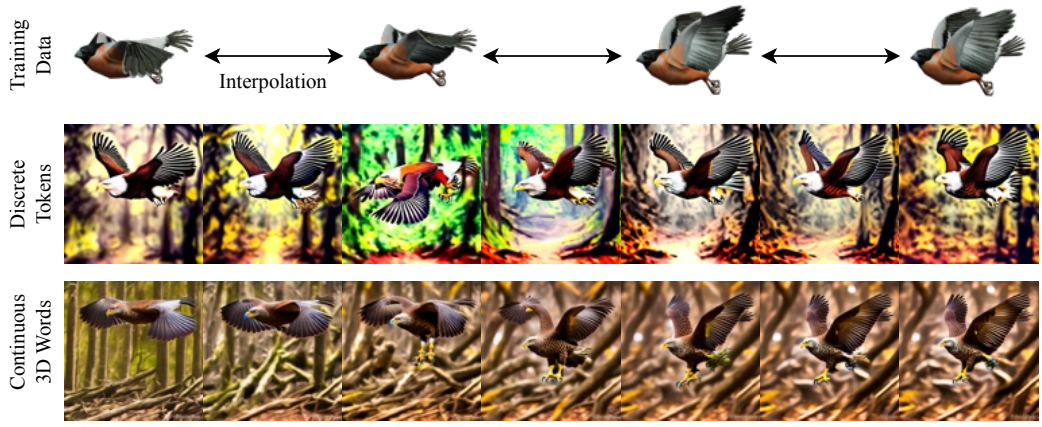

Figure 4.8: **Interpolating continuous 3D words.** We present two results of interpolation, one with discrete tokens and one with our Continuous 3D Words. Our method preserves the attributes better and enables interpolation between two values.

very hard. These benefits are illustrated in Figure 4.8. We show a comparison of learning a single *Continuous 3D Word* [☞] and 18 discrete tokens for different values of a wing pose. Both were learned with the same training method (2-stage with ControlNet augmentation). During inference, we present the generated results of three attribute values that are present in the training set, as well as the results when you interpolate the value in between. Interpolation is straightforward for our case where we just input the intermediate value into our Continuous 3D Words MLP. For discrete token, we take the interpolation of the two nearest discrete bins. Notice that the discrete tokens have difficulty not only in interpolating results but also in learning all the concepts simultaneously. Notice how the wings of the eagle in the second row of Figure 4.8 do not follow the training data and sometimes generate a completely different pose (third column, second row). On the other hand, our images closely follow the pose prescribed by the user (top row) while yielding appealing images even when the values were not seen during training (columns 2, 4 and 6; second row).

Ablation study. Figure 4.9 shows an ablation of our training strategy. We remove

Figure 4.9: **Ablation Study.** We present our ablations for: w/o two-stage training, w/o ControlNet augmentation, w/o *[obj]* as negative prompt, and our full pipeline. Notice that, when the prompt deviates a lot from the training data (truck in prompt, mesh of a dog for training), the ablated version fails to follow the prompt. Without two-stage training, it ignores the prompt and creates a dog; without the other parts it yields deformed shadows and dog-shaped trucks.

each component of our training strategy (w/o two-stage training, w/o ControlNet augmentation, w/o *[obj]* as negative prompt) and compare it with our full pipeline. Two examples are presented: one where the prompt is similar to the training and another one where it is significantly different. Without the two-stage training, the model fails to disentangle object identity with our attributes, hindering the generalization capability to new objects. This is particularly noticeable for the bottom row when the text prompt is a `truck` but a dog similar to the training mesh is generated when the two-stage training is removed. Without ControlNet, the finetuning process often overfits to the background training renderings, resulting in an inability to generate realistic backgrounds. Finally, adding *[obj]* as the negative prompt serves as a minor improvement in further disentangling both the backgrounds and object shape seen during training, resulting in a more aesthetic image.

Figure 4.10: **Non-finetuned Stable Diffusion Comparison.** We compare the generated images using the same prompts with/without finetuning the diffusion model to ensure there is no significant texture difference.

Non-finetuned Stable Diffusion. We provide 4 example comparisons using the same prompts for text-to-image generation against unfinetuned stable diffusion (Figure 4.10) to evaluate the texture differences before/after fine-tuning. While there may be a slight texture difference (potentially due to the limited single-mesh training set), we believe our newly generated image are still of high quality.

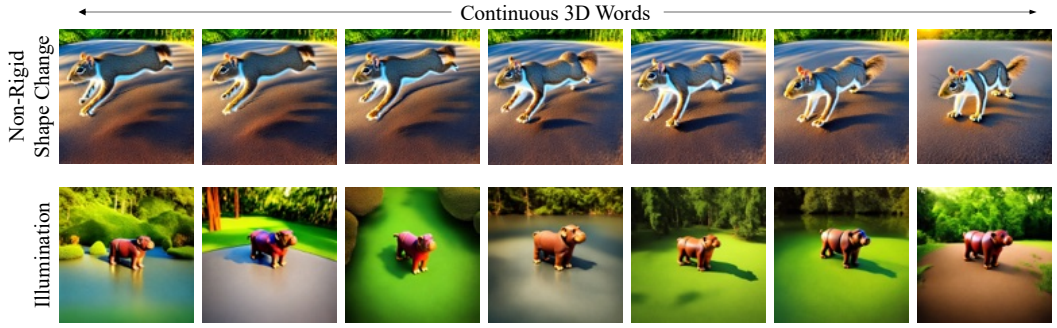

Figure 4.11: **Simpler v.s. More Challenging Attributes.** When an attribute changes the entire global scene (e.g., illumination), the results are often less fine-grained and less continuous compared to more localized changes (e.g., non-rigid shape changes).

Simpler v.s. More Challenging Attributes. While Continuous 3D Words can be used to model various types of continuous movement-based attributes, some attributes are empirically more difficult to control in a continuous manner than others. Figure 4.11 presents continuous control for illumination and non-rigid shape changes. While the time-of-day illumination changes roughly follow the guidance, the guidance is less fine-grained and changes between image to image are much more drastic compared to the changes of the raccoon. We suspect this to be due to the difference in nature of the two attributes: illumination changes the

global scene which makes it much harder to learn than non-rigid shape changes which are more regional.

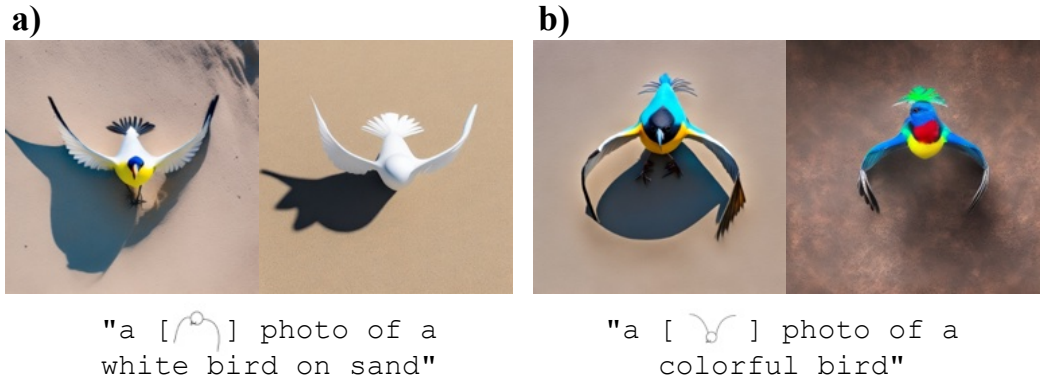

Figure 4.12: **Condition v.s. Accuracy in User Study.** a) shows two images generated by the prompt "a [ ] photo of a white bird on sand". b) shows two images generated by the prompt "a [ ] photo of a colorful bird". Left is ours, right is ControlNet (1.0). Users preferred right over left for both.

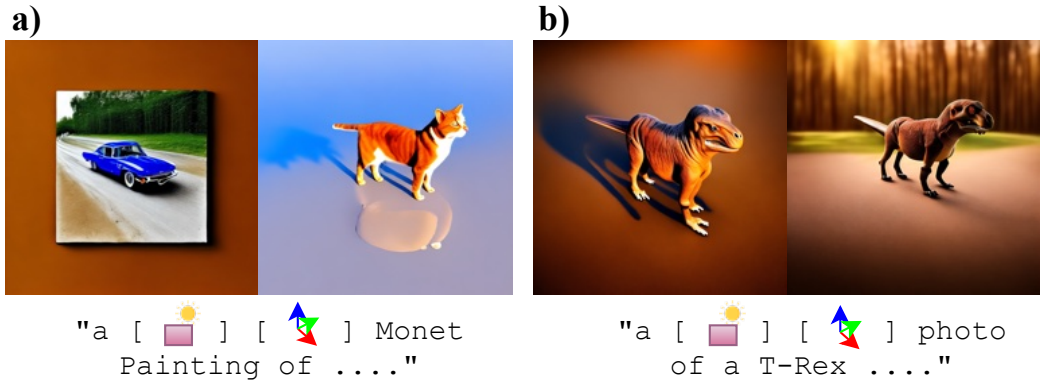

Figure 4.13: **Failure cases.** a) shows two images generated by the prompt "a [ ] [ ] Monet Painting of .....". b) shows two images generated by the prompt "a [ ] [ ] photo of a T-Rex". Both yielded suboptimal results.

Condition v.s. Generated Accuracy in User Study. During our user study, we realize that occasionally users select the image which more strictly follows the exact conditions over the image that is more physically probable. For example, in both cases a) and b) shown by Figure 4.12, the head of the bird is generated poorly, but users preferred them as they are "whiter" and "more colorful".

Failure Cases. We present in Figure 4.13 two examples of typical failure cases in our results. First, our model currently fails on more difficult scenarios where the style is given by the text prompt. In a), the image cannot fully reflect the "Monet painting" style imposed by our prompt. Second, the generated object may sometimes still overfit to the training set. In b), the T-Rex had four feet on the ground instead of standing with two claws in air – an attribute that is similar to the training dog mesh used to learn illumination.

4.6 Conclusion

We presented *Continuous 3D Words*, a framework that allows us to learn 3D-aware attributes reflected in renderings of meshes as special words, which can then be injected into the text prompts for fine-grained text-to-image generation. We made an extensive study on learning both single and multiple continuous words and show that we can control challenging attributes. With the lightweight design and promising results, we hope that this work opens up interesting applications in the vision community to create their own 3D words with a single mesh and an accessible rendering engine.

Future work. Identifying which data needs to be used for specific attributes and training multiple models for each of them is a cumbersome task. For example, some of the non-rigid motions from animals that are frequently seen in our daily lives but not greatly modeled by available animated meshes would not be disentangled with our current Continuous 3D Words setup.

On the other hand, as the amount of 3D data available significantly increases, we believe that an interesting direction is to train general models that handle multiple attributes on their own, without the need to train attribute-specific networks.

5 | Zero-Shot Material Transfer from a Single Image

The main content of this chapter is published and presented in ECCV 2024.

Figure 5.1: **Overview.** We present *ZeST*, a zero-shot single-image approach to **(a)** transfer material from an exemplar image to an object in the input image. **(b)** *ZeST*can easily be extended to perform multiple material edits in an single image, and **(c)** perform implicit lighting-aware edits on rendering of a textured mesh (see the reflection changes).

In Chapter 5, we extend beyond the motion-based concepts like like time-of-day illumination and non-rigid motions and further explore whether materials can be disentangled from object geometry and lighting given a 2D image.

We propose *ZeST*, a method for zero-shot material transfer to an object in the input image given a material exemplar image. *ZeST* leverages existing diffusion adapters to extract implicit material representation from the exemplar image. This representation is used to transfer the material using pre-trained inpainting diffusion

model on the object in the input image using depth estimates as geometry cue and grayscale object shading as illumination cues. The method works on real images without any training resulting in a zero-shot approach. Both qualitative and quantitative results on real and synthetic datasets demonstrate that *ZeST* outputs photorealistic images with transferred materials. We also show the application of *ZeST* to perform multiple edits and robust material assignment under different illuminations.

Project Page: https://ttchengab.github.io/zest

5.1 Introduction

Editing object materials in images (*e.g.,* changing a marble statue into a steel statue) is useful for several graphics and design applications such as game design, e-commerce, etc. It is a highly challenging and time-consuming task even for expert artists and graphic designers – typically requires explicit 3D geometry and illumination estimation followed by careful tuning of the target material properties (*e.g.,* metallic, roughness, transparency). Previous works try to alleviate the tedious material specification by synthesizing textures given input text prompts (Y.-Y. Yeh et al., 2024; Richardson et al., 2023). However, they are focused on texturing 3D meshes, which overlooks some of the unique challenges for material editing in 2D images, such as illumination. Another work (Sharma, Jampani, et al., 2023) proposes fine-grained material editing on images, but it cannot directly transfer materials from a given exemplar.

In this work, we aim to make 2D-to-2D material editing practical by eliminating the need for any 3D objects as well as explicit specification of material properties. Given a single image of an object and another material exemplar image, our goal is to transfer the material appearance from the exemplar to the target object directly

in 2D. See Figure 5.1 for some sample input and material exemplar images. We do not assume any access to the ground-truth 3D shapes, illumination, or even the material properties, making this problem setting practical and widely applicable for material editing.

This setup is particularly challenging from two perspectives. First, an explicit approach to material transfer requires an understanding of many object-level properties in both the exemplar and the input image, such as geometry and illumination. Subsequently, we have to disentangle the material information from these properties and apply it to the new image; the entire process has several unsolved components. Second, there currently exists no real-world datasets for supervising this task. Collecting high-quality datasets presenting the same object with multiple materials and exemplars may be quite tedious.

One of the main contributions of this work in alleviating these challenges is a zero-shot approach that can implicitly transfer arbitrary material appearances from a given 2D exemplar image onto a target 2D object image, without explicitly estimating any 3D or material properties from either image. We call our approach '*ZeST*', as it does not require multiple exemplars or any training like previous works, making it easy to generalize to any images in the wild.

With *ZeST*, we propose a carefully designed pipeline that repurposes several recent advances in 2D image generation and editing for our problem setting. At a high level, we adapt the geometry-guided generation (*e.g.,* ControlNet (L. Zhang et al., 2023)) and also exemplar-guided generation (*e.g.,* IP-Adapter (H. Ye, Zhang, Liu, Han, & Yang, 2023)) to implicitly isolate and transfer material appearance from a source exemplar to the target image while applying a foreground decolored image and inpainting for illumination cues. Our key contribution is presenting a simple pipeline with careful design choices that can be used to tackle a highly challenging problem of 2D-to-2D material transfer.

Since this is a new problem setting, we created both synthetic and real-world evaluation datasets with material exemplars and object images. Extensive qualitative and quantitative evaluations demonstrate that *ZeST* excels in photo-realism and material accuracy in the output images when compared against various baselines while being completely training-free. See Figure 5.1(a) for sample results of *ZeST*. With our pipeline, artists can grab pre-designed materials as material exemplars and directly transfer them to real-world images. By using different object masks, we can also use *ZeST*to cast different materials to multiple objects present in a single image (Figure 5.1 (b)). In addition, with slight alteration of the inputs, *ZeST* can perform light-aware material transfer by changing the reflections while keeping textural patterns consistent (Figure 5.1 (c)); this method can have potential application when used in conjunction with 3D texture generation methods (D. Z. Chen et al., 2023).

In summary, *ZeST* has several favorable properties for material editing:

- **Zero-shot, training free, single-image material transfer.** By leveraging 2D generative priors, *ZeST* works in a zero-shot manner without needing dataset finetuning. Unlike some contemporary works (Y.-Y. Yeh et al., 2024) that implicitly capture material properties using several material images, *ZeST* only needs a single material exemplar image to transfer the material in pixel space.
- **No explicit 3D, illumination or materials.** With 2D depth and segmentation estimation (which are readily available these days) and implicit material transfer, we eliminate the need for explicit specification of 3D meshes, illumination or material properties (say, in terms of BRDF).
- **Several downstream applications.** Given the simplistic and practical nature of our approach, *ZeST* can be used for several downstream graphics applications such as applying pre-designed materials to real-world images, editing multiple object materials in a single image, and perform lighting-aware

material transfer given untextured mesh renderings.

5.2 Related Work

Diffusion Models. Denoising Diffusion Probabilistic models have emerged as the state-of-the-art for class-conditional and text-prompt conditioned image generation (Dhariwal & Nichol, 2021; Ho et al., 2020, 2022; Ho & Salimans, 2022; Y. Song & Ermon, 2019; Karras, Aittala, Aila, & Laine, 2022; Kang et al., 2023). These models generate photorealistic images with exemplary geometry, materials, illumination, and scene composition. The models have been extended to be conditioned on input images for computational photography tasks such as super-resolution, style transfer, and inpainting.

Further work demonstrate controllable generation conditioned on text-based instructions (Hertz et al., 2022; Voynov, Chu, Cohen-Or, & Aberman, 2023; Ge, Park, Zhu, & Huang, 2023; M. Cao et al., 2023), semantic segmentation (Bar-Tal, Yariv, Lipman, & Dekel, 2023), bounding box (Y. Li et al., 2023; M. Chen, Laina, & Vedaldi, 2023; Z. Yang et al., 2023; X. Wang, Darrell, Rambhatla, Girdhar, & Misra, 2024), depth (Zhao et al., 2024; Bhat, Mitra, & Wonka, 2023), sketch (L. Zhang et al., 2023; Mou et al., 2023), and image prompt (H. Ye et al., 2023). Prompt-to-prompt and Prompt+ edit the input image by performing inversion followed by the introduction of new terms and reweighting the effect of terms in the input prompt (Hertz et al., 2022; Voynov et al., 2023). InstructPix2Pix performs edits an input image conditioned on an instruction (Brooks et al., 2023). Ge et al. proposed rich text based image editing allowing for style assignment and specific description to specific terms in the prompt (Ge et al., 2023). While these methods edit the image semantically and high-level descriptions, assigning specific materials using text-based approach is challenging since text acts as a limiting

modality for describing textures.

A collection of reference images can be used to learn concepts which can be further included in text prompts to generate images with the learned concepts (Ruiz et al., 2023; W. Chen et al., 2024; Kumari et al., 2023). Spatial modalities such as depth and sketches have been used for controlling the generated images (L. Zhang et al., 2023; Mou et al., 2023; H. Ye et al., 2023). Pre-trained text-to-image models can be leveraged for 3D-aware image editing using language and depth cues (Cheng et al., 2024; Michel et al., 2024; Pandey et al., 2024). The use of ControlNet has been extended by Bhat et al. to use depth for controlling the scene composition while maintaining other scene attributes (Bhat et al., 2023). Object orientation, illumination, and other object attributes can be controlled in a continuous manner using ControlNet and learned continuous tokens embedding the 3D properties (Cheng et al., 2024).

Material acquisition and editing. Material acquisition and editing is an active field of research taking into account illumination and object geometry. Previous work has demonstrated material acquisition under known illumination conditions and camera (Aittala, Weyrich, & Lehtinen, 2013; Aittala, Weyrich, Lehtinen, et al., 2015; Deschaintre, Aittala, Durand, Drettakis, & Bousseau, 2019). Such acquisition in the wild requires localizing objects with similar materials, which has been facilitated by supervised material segmentation and leveraging pre-trained vision representation backbones (Bell, Upchurch, Snavely, & Bala, 2015; Liang, Wakaki, Nobuhara, & Nishino, 2022; Upchurch & Niu, 2022; Sharma, Philip, et al., 2023). Khan et al. introduced in-image material editing using estimates of depth (Khan et al., 2006). Recent works have employed generative adversarial networks (Goodfellow et al., 2020) for perceptual material editing (Subias & Lagunas, 2023; Delanoy et al., 2022) and physical shader-based editing using text-to-image models (Sharma, Jampani, et al., 2023). The use of generative models has

been extended to explicitly learning materials (Lopes et al., 2023) and texturing 3D meshes (Y.-Y. Yeh et al., 2024; D. Z. Chen et al., 2023; Richardson et al., 2023; T. Cao et al., 2023).

In our work, we aim to use pre-trained image generation diffusion models to perform exemplar-based material transfer from a single image. We aim to use ControlNet and IP-adapter to perform material transfer in a zero-shot way without any training.

5.3 Method

In this section, we describe our method *ZeST* that performs exemplar-based material transfer. Recent methods perform the related problem of texture synthesis on meshes (Richardson et al., 2023; Y.-Y. Yeh et al., 2024) by finetuning a diffusion model on 3-5 material exemplar images to capture the texture/material in the latent space. On the contrary, *ZeST* only requires a single material exemplar image and a single input image, accomplishing material transfer in a zero-shot, training-free manner.

5.3.1 Problem Setting

Given a material exemplar image M and an input image I, we aim to output an edited image I_{gen} from I by transferring the material from the material exemplar to the object in the input image while preserving other object and scene properties (e.g. object geometry, background, lighting etc.). Performing this task requires understanding the material, geometry, and illumination from both the exemplar and the input image.

In practice, estimating all the aforementioned object-level properties and further isolating material information explicitly from M is challenging since these proper-

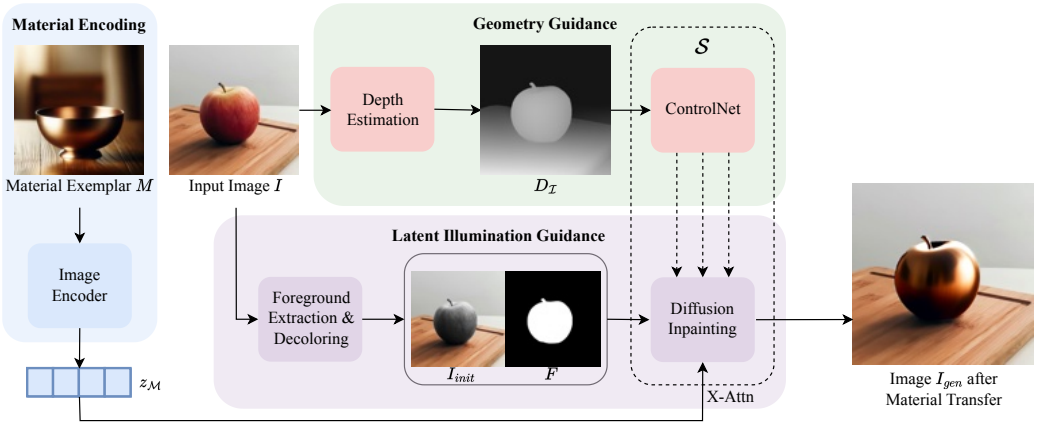

Figure 5.2: ***ZeST* Architecture.** Given a material exemplar M and an input image I, we first encode material exemplar with an image encoder (*e.g.,* IP-Adaptor). Concurrently, we convert the input image into a depth map D_I and a foreground-grayscaled image $I_{init}$to feed into the geometry and latent illumination guidance branch, respectively. By combining the two sources of guidance with the latent features from the material encoding, *ZeST* can transfer the material properties onto the object in input image while preserving all other attributes.

ties are entangled in the pixel space. Therefore, we propose to tackle this problem in the latent space of diffusion models. Specifically, we aim to extract a latent representation z_M containing the material and texture information that we can then inject into a generative diffusion model $\mathcal{S}$ to generate I_{gen}.

5.3.2 *ZeST* Overview

Since there currently exists no synthetic/real image dataset to supervise the learning of a 2D-to-2D material transfer, we perform the material transfer in a zero-shot training-free manner. Therefore, we first break down this complex task into sub-problems of (1) encoding the material exemplar, (2) geometry-guided image editing, and (3) making the generation process illumination-aware. Given the recent advances in high-fidelity diffusion models and complementary adapters for image generation, we leverage existing pre-trained modules to tackle each of the sub-problems that together compose our pipeline to perform image-prompted

material editing.

Figure 5.2 presents an overview of our pipeline, which comprises three branches to guide the material, geometry, and lighting information, respectively. The Material Encoding branch takes the material exemplar image M as input, which is processed by the image encoder to obtain a material latent representation z_M.

Concurrently, we feed the input image I into Geometry Guidance and Latent Illumination Guidance Branch. The Geometry Guidance branch computes the depth map D_I for the image I, which is used as the input to ControlNet. The Latent Illumination Guidance branch computes a foreground mask F using I and creates a foreground-grayscale image I_{init}, which we use as input to the Diffusion Inpainting pipeline. We concatenate the embeddings from ControlNet with the inpainting diffusion model at the corresponding and inject the material embedding z_M through the cross-attention. The output of the inpainting diffusion model, I_{gen}, with the edited image containing the object in I cast with material from exemplar image M.

Our design choices to facilitate computation of material embedding, geometry guidance, and illumination cues are discussed in the following sections.

5.3.3 Encoding Material Exemplar

Given the material exemplar image M, this branch encodes the image into a latent representation while preserving its material properties. Previous works (Richardson et al., 2023; Y.-Y. Yeh et al., 2024) address this by finetuning a text-to-image diffusion model to encode the image into a rare token, implicitly treating the rare token as a latent representation that can be used in conjunction with other texts for image generation. However, this approach of optimizing for the material token requires the time-consuming step for every new material exemplar and usually requires 3-5 images to prevent overfitting.

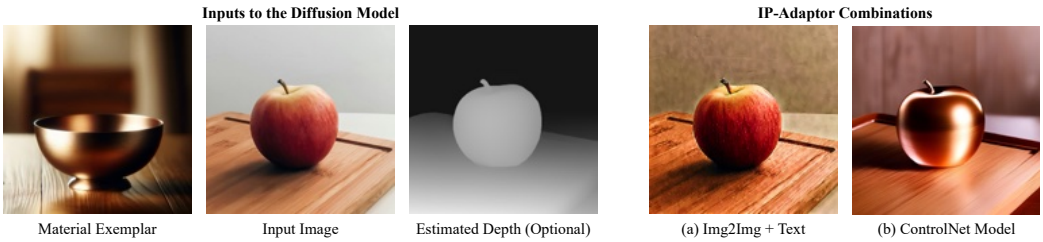

Figure 5.3: **The design choice of IP-Adaptor with ControlNet**. Given the material exemplar and the input image, we dive into the different choices of utilizing the IP-Adaptor. In particular we realize that an Img2Img + text module (a) wouldn't properly transfer the materials properly to the main object. On the other hand, ControlNet (b) will preserve the geometry information of the given input. We thus utilize this as the starting point for geometry guidance to further explore the best illumination cues.

We draw inspiration from the recently introduced IP-Adapter (H. Ye et al., 2023). The IP adapter uses a CLIP image encoder to extract image features that can be injected into a diffusion model via the cross-attention layers. These features can be used as an additional condition to guide text prompts or other mediums for the generation. For example, one can input an image of a person and then describe "on the mountain" with text to obtain an image of the person in the mountains.
However, we realize that IP-Adaptor does not work well when combined with an Img2Img pipeline, as shown in Figure 5.3 (a) for our task. Moreover, adding text guidances like "changing the apple texture to golden bowl" does not produce photorealistic output and does not preserve other scene information (*i.e.* background). This problem of geometry and material entanglement within material embedding z_M remains unsolved, thus motivating the need for geometry and illumination guidance.

5.3.4 Geometry Guidance via Depth Estimation

Since decoupling geometry and material properties in images is challenging and requires additional training data, we provide an alternative solution where we

enforce a stronger geometry prior to the diffusion model to overwrite the structural information present in z_M. To this end, we adopt a depth-based ControlNet to provide geometry guidance from the input image I. We observe that the geometry information from the depth map D_I overwrites the geometry information encoded in the z_M (see Figure 5.3 (b)). Note that with the geometry enforced by using depth-based ControlNet, we can successfully transfer the golden material of the bowl to the apple.

While the use of ControlNet with IP-Adaptor is introduced in the original IP-Adaptor paper (H. Ye et al., 2023), we employ it for a different purpose of applying new structural control over an object in the image (*e.g.,* changing a person's pose). After extensively comparing various components for encoding the material exemplar and input image (analysis in Section 5.4.2), we find the depth-based guidance from pre-trained ControlNet helps us preserve the original geometry of the object for the task of material transfer.

While the addition of ControlNet helps preserve the geometry, we observe that the results suffer from inconsistency in preserving the illumination and background from the input image. This is evident in Figure 5.3, where the background and the lighting changes differ from the input.

5.3.5 Latent-Space Illumination Guidance

Our final branch is primarily responsible for preserving the illumination and background in the input image. We propose two-fold guidance for illumination in the latent space during generation – an inpainting module and a foreground decoloring process. In addition to the attached IP-Adaptor and ControlNet, we adopt an inpainting diffusion model $\mathcal{S}$ instead of a standard generator. Specifically,

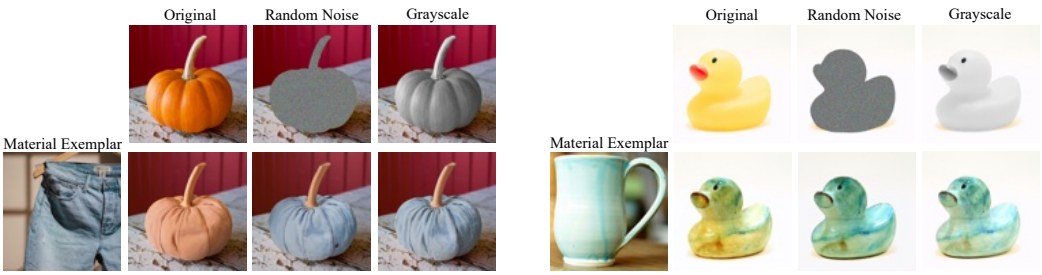

Figure 5.4: **Ablating input for illumination guidance.** To validate our design choice of the foreground-grayscale image for initializing inpainting, we compare the generated results against using the original image and random noise as inputs. The original image presents a strong base color prior that perturbs the generation, while the random image neglects shading information, leading to wrong lighting in both examples.

our ControlNet-inpainting procedure takes in four conditions for image generation:

$$I_{gen} = \mathcal{S}(z_M, D_I, I_{init}, F), \tag{5.1}$$

where z_M is the material encoding, D_I is the depth map computed for input image I, I_{init} is the initial image to denoise from, and F is the foregound mask of target object in I which we are editing.

We conduct an ablation on the various versions of I_{init}, as shown in Figure 5.4. Specifically, we test out the following settings: (1) using the original input image, (2) initializing the foreground with random noise, and (3) using the foreground grayscaled image. Intuitively, directly letting $I_{init} = I$ (Setting (1)) would be a preferable option as I encompasses implicit lighting information (from the object's shading and the surrounding environment) while conveniently enforces all other parts of the image other than the object to remain the same. In practice, however, we found that using the original image inevitably introduces a strong prior of the base color from the input object (e.g. orange color of pumpkin), which would be entangled with the material base color from M in the output image. This artifact is sustained even when we significantly extend the number of denoising

steps. On the other hand, when initializing I_{init} with random noise, the method indeed removes the base color prior but also removes the shading information causing incorrect illuminations in the synthesized object (e.g., the left side of the synthesized pumpkin is darker, but light is coming from the left). In our proposed pipeline, we perform grayscale operations in the pixel space for the object region (3). This provides a balanced solution of removing the strong color priors from the input image while keeping the shading cues for the inpainting diffusion model. Thus, we propose to initialize I_{init} as:

$$I_{init} = F \odot I_{gray} + (1 - F) \odot I, \tag{5.2}$$

which is converting the appearance of foreground object in the image to grayscale. By doing so, $(1 - F) \odot I$ implicitly preserves the lighting direction, intensity, and color information, while $F \odot I_{gray}$ preserves the shading information of the object without any base color prior.

5.3.6 Implementation Details

We implement our method using Stable Diffusion XL Inpainting (Podell et al., 2023) with the corresponding version of depth-based ControlNet (L. Zhang et al., 2023) and IP-Adaptor (H. Ye et al., 2023). We use Dense Prediction Transformers for depth estimation (Ranftl, Bochkovskiy, & Koltun, 2021) and Rembg[1] for foreground extraction. Our method is implemented in PyTorch and runs on a single Nvidia A-10 GPU with 24 GB of RAM. For all Dreambooth approaches, we use the official LoRA-Dreambooth provided by Diffusers.

[1] https://github.com/danielgatis/rembg

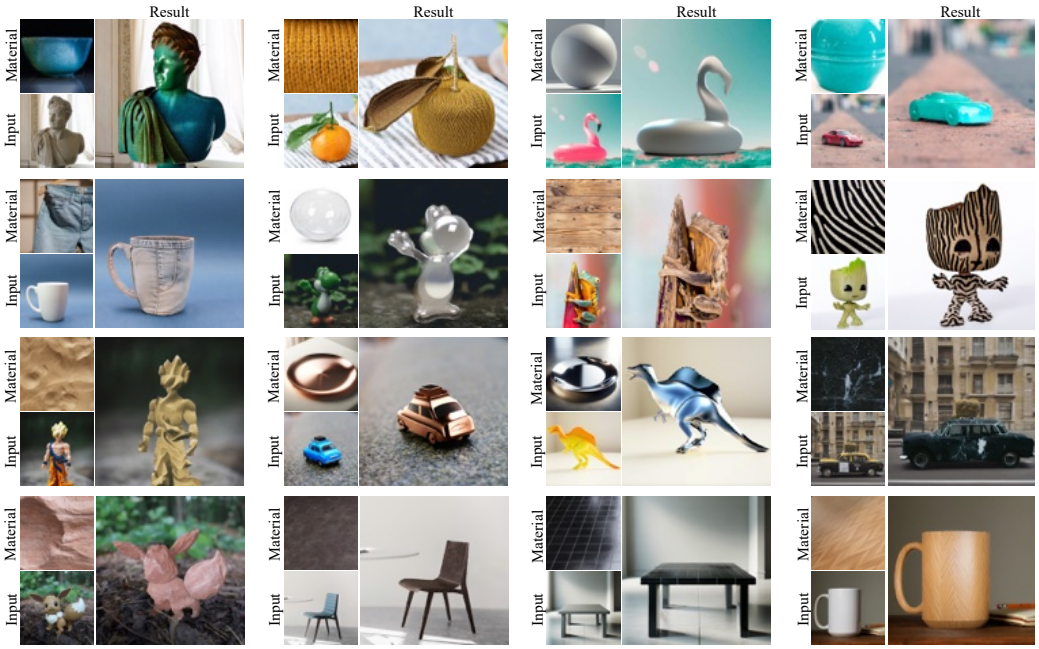

Figure 5.5: **Qualitative results on diverse materials.** We present results of material transfer from a diverse set of material exemplar images. Even when perturbed by lighting and complex geometry, *ZeST*can still isolate the material information from the exemplar image and transfer to various objects while preserving the original geometry and illumination conditions. Note the change in specular regions as shinier materials are chosen in the case of the car made of brass and the dinosaur made of shiny steel.

5.4 Experiments

We evaluate the efficacy of our method against various baselines. We also present several examples of downstream applications using our method.

5.4.1 Datasets

As the first to propose this problem, we create two datasets for comparison and evaluation. The real-world datasets provide us an understanding of our model's robustness, while the synthetic dataset is used for standard quantitative metrics.

Real-World Dataset. We curate a dataset comprising of 30 diverse material

exemplars and 30 input images, collected from copyright-free image sources (*i.e.* Unsplash) and images generated by DALLE-3. All of these images are object-centric, where there exists a main object in the foreground to which we are extracting the material from or applying the material onto.

Synthetic Dataset. To perform quantitative evaluation, we use Blender to create a synthesized dataset of 9 materials randomly initialized by adjusting the base color, metallic, and roughness, and 20 meshes of different categories from Objaverse (Deitke et al., 2023) rendered in three random viewpoints each, generating 540 ground-truth renderings. We render spheres assigned with each material individually and use the rendered image the material exemplar and pre-textured mesh rendering as input for all methods.

While *ZeST* is completely training-free, other methods of learning materials (e.g., Dreambooth) require further fine-tuning for every exemplar given. This makes it infeasible to scale up the two datasets. Both our datasets are of comparable sizes to previous works on finetuning diffusion models (Y.-Y. Yeh et al., 2024; Ruiz et al., 2023).

5.4.2 Qualitative Results

Material transfer results on real images. To demonstrate the application of *ZeST* on a wide range of materials and objects, we present examples of material transfer in Figure 5.5. The first three rows present results on real-world images, while the fourth row shows results using PBR materials [2]. Based on the examples, we observe that the material is properly disentangled from the geometry in the material exemplar and follows the shape of the object in the input image. This is particularly evident in the results of the orange, frog, and Groot toy figure, where the material is completely flat. We also notice accurate shadings in the bust and

[2] https://www.textures.com/browse/pbr-materials/114558

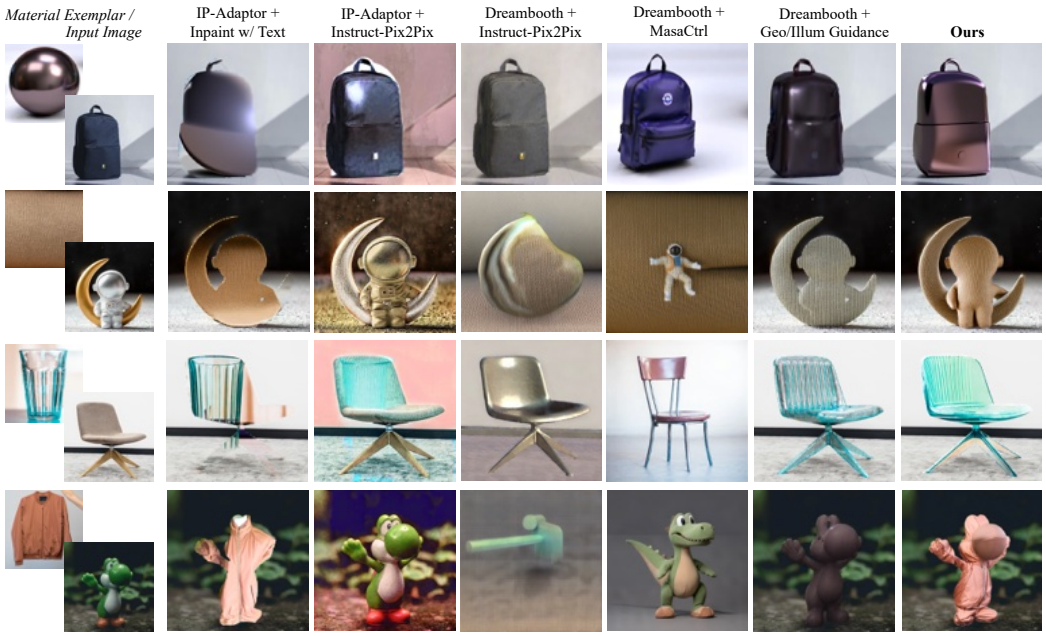

Figure 5.6: **Qualitative comparisons against baselines.** Given the material exemplar and input image in the first column, we compare our method to five different baselines. Without any geometry guidance, all image editing baselines fail to impose the correct geometry of the input image. On the other hand, using Dreambooth with our geometry and illumination guidance often contains albedo shifts, potentially due to information loss when encoding material properties into a word token.

table examples when comparing them against their inputs. In the car and toy dinosaur examples, the reflections from the exemplars are isolated from the textural patterns and cast reasonably based on the illumination cues.

Qualitative comparisons. Since our work is the first to perform material transfer in latent space, we modified existing methods to compare against. Specifically, since existing image-guided texture synthesis methods utilize Dreambooth for their first step to encode the textures from images into word tokens (Corneanu, Gadde, & Martinez, 2024; Richardson et al., 2023; Y.-Y. Yeh et al., 2024), we set Dreambooth as the backbone for learning material properties and combine with text-guided image editing techniques for comparison, including MasaCtrl and Instruct-Pix2Pix, and using *ZeST* but swapping out the IP-Adaptor with text. While

our method is training-free, Dreambooth requires finetuning for every material exemplar given. We also explore alternative options to combine with IP-Adaptor, including text-guided inpainting and Instruct-Pix2Pix with the prompt "Change the texture of the object".

We present qualitative comparisons against the baselines on four exemplar and input images in Figure 5.6. By using Inpainting with Text prompt instead of ControlNet, the model ignores the geometry of the original input when casting the materials. In both cases when using Instruct-Pix2Pix (with IP-Adaptor or Dreambooth), the geometry of all objects is better preserved, but the model fails to capture the material property from the material exemplar image. The combination of Dreambooth and MasaCtrl fails to preserve the geometry of the object in the input image and misattributes the material. The closest baseline to ours is Dreambooth with our proposed geometry and illumination guidance; however, we observe that the word encoding process results in some information loss as evident in the color shifts of the backpack and the astronaut figure. Furthermore, the method requires additional training for every material exemplar, whereas *ZeST* takes roughly 15 seconds to generate the image.

Our method, *ZeST*, performs the task effectively by retaining the object geometry, scene illumination, and attributing the material correctly. Additionally, note that *ZeST* adapts to more challenging material exemplar images, such as transparent materials (glass cup in Figure 5.6 Row 3) and images with other minor objects (additional hand in Figure 5.6 Row 4).

5.4.3 Quantitative Comparisons

We follow previous work (Sharma, Jampani, et al., 2023; Y.-Y. Yeh et al., 2024) and use the synthetic images to compare all methods in terms of PSNR, LPIPS (R. Zhang, Isola, Efros, Shechtman, & Wang, 2018), and CLIP similarity

Table 5.1: **Quantitative Comparisons and User Study**. We grab the strongest baselines in our qualitative comparisons for additional studies. Left: We measure the PSNR, LPIPS (R. Zhang et al., 2018), CLIP similarity score (Radford et al., 2021), and DreamSim (Fu et al., 2024) in a quantitative study on the synthetic dataset. Right: We perform a user study to evaluate the material fidelity and photorealism of the edited images from each method. We randomly sample 5 out of 900 real-world exemplar-input combinations for each of the 16 participants.

	PSNR↑	LPIPS↓	CLIP↑	DreamSim↓
IP-Adaptor + Instruct-Pix2Pix	17.08	0.099	0.740	0.390
DB + Our Geo/Illum. Guidance	25.52	0.058	0.874	0.238
Ours	**25.59**	**0.053**	**0.883**	**0.198**

	Fidelity↑	Photorealism↑
IP-Adaptor + Instruct-Pix2Pix	1.48	3.23
DB + Our Geo/Illum. Guidance	3.25	3.41
Ours	**4.05**	**3.78**

score (Radford et al., 2021) against ground truth renderings. We also incorporate another DreamSim (Fu et al., 2024), a more recent metric that is more similar to human references. We use IP-Adaptor + Instruct-Pix2Pix and Dreambooth + our geometry and illumination guidance as baselines, as they are the strongest (and only) performers from our qualitative comparisons that can roughly edit the material based on the geometry.

Table 5.1 (left) presents our results. We see a dramatic improvement when shifting from the instruct-pix2pix pipeline to our geometry and illumination guidance. While using Dreambooth performs similarly to our IP-Adaptor in the synthetic dataset, it requires a fine-tuned model for each material exemplar, making it unfeasible to scale up. In addition, we show in the next section that our method excels in real-world datasets.

User Study. We also create a user study with 16 participants to understand the capability of our model given real-world materials tested on real images. Each subject is shown 5 random samples from the 900 combinations generated from the dataset with our method and against the two strongest baselines: Dreambooth + ControlNet-Inpainting and IP-Adaptor + Instruct-Pix2Pix. We ask each subject to rate each image from 1 to 5 based on (1) material fidelity: how close the material in the generated image is compared to the original exemplar and (2) photorealism: how realistic the generated image is. Our results are summarized in Table 5.1

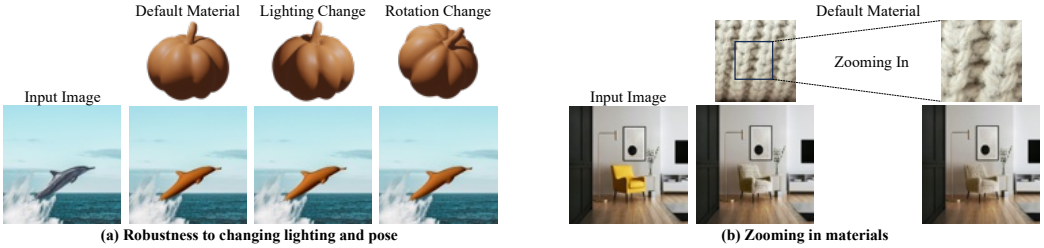

Figure 5.7: **Robustness to lighting and object pose.** We present two types of robustness testing. **(a)**: Robustness to changing the material exemplar lighting and pose. **(b)**: Zooming into the material exemplar. Our model yields highly similar results in both, showing the capability to adapt to these external changes.

(right).

Our results show significant improvements from the two baselines in both material fidelity and photorealism of the edited image. The score improvements are also greater in real-world scenarios compared to synthetic ones. This could be the result of information loss during finetuning and overfitting to the exemplar background, which is less significant under controlled synthetic scenarios.

5.4.4 Robustness of the Model

In addition to the diverse set of results presented in Figure 5.5, we extensively test out the behavior of *ZeST* with special cases of material exemplar images.

Relighting and rotating the object in the material exemplar image. A good material extractor should be agnostic to small lighting and rotation changes of the same object used as the material exemplar. To evaluate this, we render a random material and cast it onto an irregular-shaped pumpkin (another example is in the Appendix). We then render three samples of the pumpkin, a default lighting orientation, a change in lighting direction pitch by 120 degrees, and a random rotation, as shown in 5.7 (a). The transferred materials onto the dolphin remain roughly consistent across all samples, showing that our method is fairly resistant to

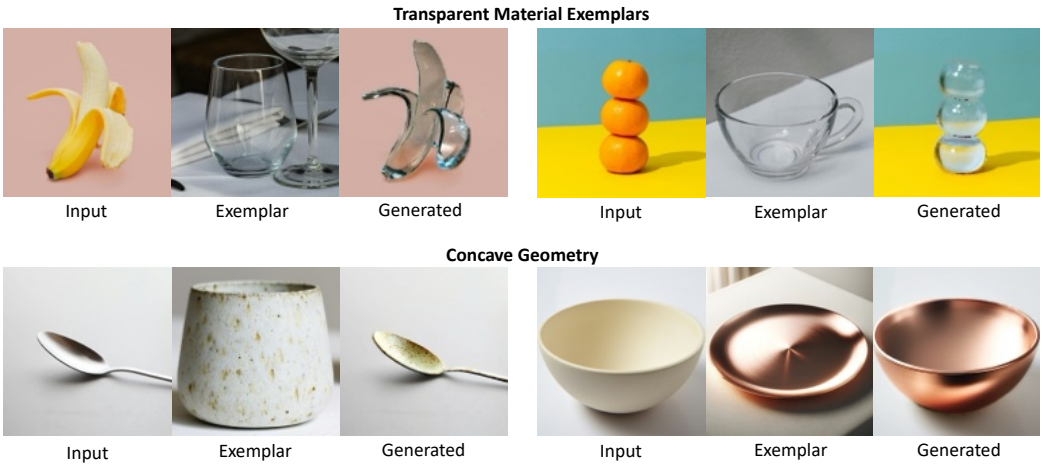

Figure 5.8: **Results on Transparent materials and concave geometry.** We show results of transparent material exemplars (Row 1) and concave input images (Row 2).

these changes at a small scale.

Effect of image scale of material exemplar image. To examine the effect of the scale of the material exemplar, we first use an image of a woolen cloth material with a distinctive repeating pattern and apply our method to an image of a chair. Then, we zoom into the exemplar image manually to the edge only very few repeated patterns are left. Our results in Figure 5.7 (b) show that while the scale of the material is drastically different, the model automatically re-adjusts the patterns into a reasonable size to be cast onto the input image.

Transparent material exemplars. To further explore the capabilities of *ZeST* on optically complex material exemplars, we show two examples (Figure 5.8, Row 1) on transferring transparent materials to the new object. Despite the glass material being fully see-through, the model picks up the transparent property and reflects it accordingly to the new scene presented in the input image.

Material editing on concave geometry. We also show two examples in Figure 5.8, Row 2, on transferring materials to objects with concave geometry. Concavity leads to suddenly more abrupt shadow changes and therefore is are more difficult to be solved with extrinsic image decompositions. However, our model still provides

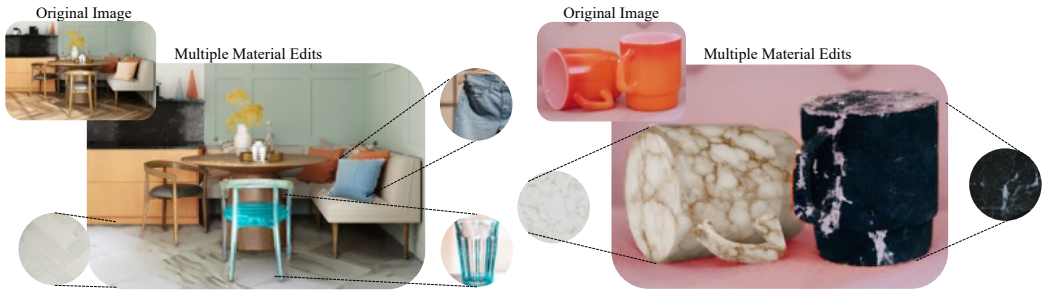

Figure 5.9: **Multiple Material Transfers in a Single Image.** By replacing the foreground extraction with an open-vocabulary segmentation module (*e.g.,* SAM) to obtain multiple masks, *ZeST*can be applied iteratively to cast different material properties to different objects in a single RGB image.

visually convincing results in both of the examples provided.

5.4.5 Applications

Applying multiple materials to multiple objects.

By replacing the foreground extraction with an open-vocabulary segmentation module (*e.g.,* SAM) to obtain multiple masks, *ZeST* can be used to iteratively change multiple materials in a single RGB image. Figure 5.9 presents two examples of editing multiple objects in a single image. As evident in the transparent glass chair where the wooden table behind is roughly visible, *ZeST* can generalize complex scenes with multiple objects. Note that the order of multiple object edits also matters. In particular, if the material of one object is reflective of the other, it would be advisable to apply this material at the latest so that the reflections take into account editing changes already made.

Exemplar-based 3D Texturing. *ZeST* can also be amended to apply textures onto 3D meshes. Recently proposed text-driven texturing pipeline – Text2Tex (D. Z. Chen et al., 2023) – uses 2D text-to-image Stable Diffusion and ControlNet to render and refine mesh texture on a per-view basis with a text prompt. We can replace this texturing backbone of Text2Tex directly with *ZeST* and remove the

Figure 5.10: **Exemplar-prompted 3D Texturing.** We can also combine *ZeST* easily with existing text-driven texturing techniques (D. Z. Chen et al., 2023). We show examples from two meshes, each using to two material exemplars.

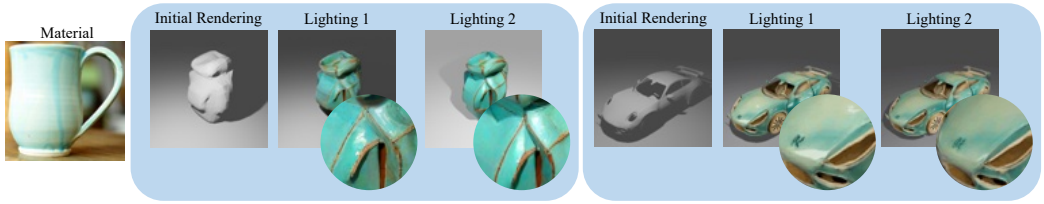

Figure 5.11: **Lighting-aware Image Editing.** Given a rendering of a untextured mesh, we can alter *ZeST* slightly to achieve lighting-aware material edit. It can be seen from both examples where the reflection can be disentangled from the object texture.

illumination guidance branch, enabling mesh texturing with a material exemplar image. Figure 5.10 presents four examples of exemplar-based texturing (two from a Porsche mesh and two from a backpack mesh). Even when the exemplar varies from the original geometry (see Row 2 for both cases), the model learns to apply the textures appropriately based on the geometry. We also realize that providing text description (*e.g.,* back view of a backpack) in addition to the material exemplar is particularly helpful in making the texture consistent across all views. Note that existing methods for exemplar-based mesh texturing (Richardson et al., 2023) converts the exemplar image(s) into words via Dreambooth (Ruiz et al., 2023) before texturing. Using *ZeST* makes the texturing process much faster and more scalable.

Lighting-aware Material Transfer. Given a material exemplar image and an untextured mesh rendered under multiple illumination conditions, *ZeST* can also

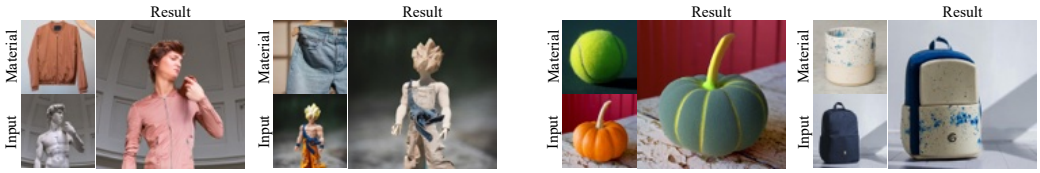

Figure 5.12: **Limitations.** Our method primarily fails in two modes. **(a)** Even when the ControlNet-Inpainting is enforced upon the entire object, the model sometimes picks the most "probable" areas to transfer the material, instead of casting the material on the entire object. **(b)** If two textures are present in the exemplar image (e.g., foreground and background of the tennis ball, the glazed top and bottom logo of the cup), the model sometimes combine both materials when performing the edit.

be used to perform lighting-aware material transfer. Specifically, we first generate the materials and textures of the image under Lighting 1 using *ZeST*. Then, by fixing the same seed during generation and using the generating image given the first lighting as the input to the second, we can enforce consistency in the material and texture generated (details of implementation in Appendix), but change the reflections based on the latent space understanding of the material exemplar. We show examples of transferring the glazed cup material to two mesh renders in Figure 5.11. *ZeST* successfully disentangles the reflections while keeping most textural patterns consistent between the two images. This technique could potentially be applied jointly with other 3D texture synthesis works (D. Z. Chen et al., 2023).

5.4.6 Limitations

While our method demonstrates generalizable results for the task of single-image exemplar-based material transfer, it still encompasses several limitations. Since we operate majorly in the latent space, the model sometimes exhibits uncontrollable behaviors based on its image understanding. Figure 5.12 presents two forms of more frequent failure cases: (a) Partial material transfer: the material is only transferred to parts instead of the entirety of the object. We hypothesize that the

failure stems from the entanglement of material properties and the exemplar's identity, as the material is only applied to where it seems the most probable (*e.g.,* only apply the jacket material to the statue's body). (b) Blending multiple materials: since the current IP-Adaptor does not have a module to extract regions of an image for material transfer, *ZeST* sometimes mixes up multiple materials in the exemplar image during transfer.

5.5 Conclusion

We present *ZeST*, a zero-shot, training-free method for exemplar-based material-editing. *ZeST* is built completely using readily available pre-trained models and demonstrates generalizable and robust results on real images. We curate synthetic and real image datasets to evaluate the performance of our approach. We also demonstrate downstream applications like multiple edits in a single image and material-aware relighting. *ZeST* serves as a strong starting point for future research in image-to-image material transfer, implying opportunities of leveraging pre-trained image diffusion models for complex graphic designing tasks.

6 | Conclusion and Future Prospects

In conclusion, this thesis explores three research challenges in extracting a wide range of 3D information (geometry, continuous movement-based attributes, materials) from models trained with large-image datasets with limited availability of 3D data. Each of the pipelines utilizes different sets of data or 2D models and hence provides different insights. We present the contributions and implications for each chapter as well as potential future work in learning and controlling 3D information.

6.1 Summary of Contributions and Lessons Learnt

In Chapter 3, we present 3DMiner, an end-to-end pipeline to divide large image collections into geometry-consistent clusters, find coarse keypoint correspondences, and reconstruct an occupancy field using bundle adjustment, all without the need for 3D ground truths. The exploration shows that 2D image datasets already contain geometry information that can be easily distilled via training-free augmentations and clustering. While the shapes mined out are often coarse, 3DMiner serves as a solid starting point for a new problem formulation of mining 3D shapes from web-retrieved images without any priors or annotations. With improvements from each of the plug-and-play modules, such a pipeline could be very useful in continuously discovering 3D geometries without much human effort.

In Chapter 4, we extend beyond geometry and into extracting and controlling continuous movement-based attributes, such as time-of-day illumination and non-rigid motion. We present Continuous 3D Words, a way to encode these attributes into sliders that can be used jointly with texts. With a series of training strategies to disentangle object attributes from object identity, continuous 3D Words can be trained with as few as one single animated mesh. This phenomenon provides three interesting insights. First, 3D awareness is likely already disentangled within the

image features despite not being described by texts, and therefore our training requires only very little data to map the sliders to particular concepts. Second, despite the domain gap, large pretrained diffusion models encompass the capabilities to transfer attributes learnt from synthetic datasets to real-world images, allowing interesting applications in 3D-aware image editing. Finally, these 3D-aware concepts can be combined seamlessly with other forms of controls to unlock various flexible controls for graphic design tasks.

Finally, we explore material extractions and editing in Chapter 5. We present *ZeST*, a zero-shot approach to transfer materials from one image exemplar to another. Unlike prior works that operate by decomposing the image into explicit 3D, illumination, or materials, *ZeST* utilizes a range of deep-learning-based 2D techniques to implicitly transfer the material features in the latent space. The training-free nature of our pipeline is a crucial indicator showing that complex properties like materials are actually disentangled and within the latent features of image-pretrained models, which aligns closely with the implications we reached in Chapter 4. With careful design, one can isolate just material appearances from shading and geometry of objects within an image without requiring a designated dataset for finetuning. More importantly, our zero-shot approach also shows that many attributes (3D or not) could already exist in the current plethora of readily available generative models for us to discover.

All of these directions point to an interesting outlook of transforming generative models into flexible renderers. Traditionally, to use physical rendering engines to render images, one would need to carefully define the geometry, illumination, and materials in order to obtain a semi-realistic image to a user's liking. While the controllability is high, one is constrained by the available data, making the design space rather limited. With generative models, all of these inputs turn into optional controls. We can have the benefits of both allowing generative models to fill in

the blanks, but also gaining fine-grained controls over small details towards image composition. Incorporating generative models into graphics design pipelines could be very beneficial to improving the efficiency of designing various visual arts and effects.

6.2 Limitations of Current Work

While each of the works in this space is a significant milestone towards learning and controlling 3D information, there are still important open problems in this research domain. These research challenges can be roughly separated into two categories: 1) better modeling of 3D-aware information and 2) better control over 3D-aware information.

6.2.1 Better Modeling of 3D-Aware Information

- Recovering better geometry – The current 3DMiner builds from the hypothesis that all the images within the cluster are of very similar/almost the same shape. This hypothesis creates two potential challenges. First, when one or a few of the images deviate slightly from others, the shape becomes coarse and suboptimal for any downstream tasks. One potential idea is to use the current geometry as a coarse prior, in which one can find ways to further refine it to fit one particular image's geometry. Second, many clusters contain too few images to obtain reconstructions with high fidelity. With the plug-and-play nature of 3DMiner, future works could potentially seek to improve upon individual components such as feature extraction by experimenting with better models (e.g., DiNOv2 (Oquab et al., 2023)). Several works after 3DMiner, such as ReconFusion (combining synthetic and real-world views) and CAT3D (single image to 3D) are also promising directions of further

incorporating priors from large image datasets to obtain better reconstruction quality (R. Wu et al., 2023; Gao* et al., 2024).

- Better synergy between material and lighting for multi-view consistency – While Continuous 3D Words and *ZeST* can model time-of-day illuminations and materials in a single image, they are not multi-view consistent. This is because the modeling of lighting and material is an ill-posed question under a single image – while two generated images can be of the same material and the same lighting directions viewed from two angles, attributes such as reflections may not be consistent with one another. Future controls to ensure better multi-view consistency under various lighting remain a challenging and interesting direction to explore.

6.2.2 Better Control over 3D-Aware Information

- Create a plausible control space depending on the image context – While works like Continuous 3D Words, *ZeST* and many other concurrent works (M. Chen et al., 2023; Pandey et al., 2024) create various types of controls over synthetic and real-world images, these controls are not and shouldn't be universal. For example, a wing-pose controller should not be available to a photograph of a dog; a time-of-day illumination slider should not happen inside a room without windows. Being able to understand the image and infer a set of plausible 3D-aware controls is a hugely challenging and open task.
- Controlling spatially-varying materials – Finally, the controllability of spatially-varying materials is also an interesting direction to move forward. Properties such as the size and rotation of the pattern are likely to be encoded in the features of the noise of diffusion models, and not within the CLIP space where our work *ZeST* controls the materials.

6.3 End Note

The core objective of computer vision is to let computers understand and interpret visual data of the real world, and the capability to reason in 3D is paramount to various aspects of this research area, both in generative and discriminative tasks. 2D datasets, sourced from photos, videos, paintings, and art pieces, are rich and diverse assets that are likely to still be much more abundant compared to 3D datasets for a long time. It therefore is very important to be able to extract and control 3D information with pretrained 2D models and limited guidance from small 3D datasets, as it would bring a plethora of applications in both creating new 3D assets and allowing better 3D-aware controls over generative models.

Bibliography

Agarwal, S., Snavely, N., Simon, I., Sietz, S. M., & Szeliski, R. (2009, September). Building rome in a day. In *Iccv.*

Aittala, M., Weyrich, T., & Lehtinen, J. (2013). Practical svbrdf capture in the frequency domain. *ACM Trans. Graph.*, *32*(4), 110–1.

Aittala, M., Weyrich, T., Lehtinen, J., et al. (2015). Two-shot svbrdf capture for stationary materials. *ACM Trans. Graph.*, *34*(4), 110–1.

Alaluf, Y., Richardson, E., Metzer, G., & Cohen-Or, D. (2023). A neural space-time representation for text-to-image personalization. *arXiv preprint arXiv:2305.15391*.

Amir, S., Gandelsman, Y., Bagon, S., & Dekel, T. (2021). Deep vit features as dense visual descriptors. *arXiv preprint arXiv:2112.05814*.

Bain, M., Nagrani, A., Varol, G., & Zisserman, A. (2021). Frozen in time: A joint video and image encoder for end-to-end retrieval. In *Ieee international conference on computer vision.*

Bar-Tal, O., Yariv, L., Lipman, Y., & Dekel, T. (2023). Multidiffusion: Fusing diffusion paths for controlled image generation.

Bell, S., Upchurch, P., Snavely, N., & Bala, K. (2015). Material recognition in the wild with the materials in context database. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 3479–3487).

Betker, J., Goh, G., Jing, L., TimBrooks, â., Wang, J., Li, L., ... Ramesh, A. (2023). *Improving image generation with better captions.* Retrieved from https://api.semanticscholar.org/CorpusID:264403242

Bhat, S. F., Mitra, N. J., & Wonka, P. (2023). Loosecontrol: Lifting controlnet for generalized depth conditioning. *arXiv preprint arXiv:2312.03079*.

Blattmann, A., Dockhorn, T., Kulal, S., Mendelevitch, D., Kilian, M., Lorenz, D.,

... others (2023). Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*.

Bolles, R. C., & Fischler, M. A. (1981). A ransac-based approach to model fitting and its application to finding cylinders in range data. In *Proceedings of the 7th international joint conference on artificial intelligence - volume 2*. Morgan Kaufmann Publishers Inc.

Brooks, T., Holynski, A., & Efros, A. A. (2023). Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 18392–18402).

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., ... others (2020). Language models are few-shot learners. *Advances in neural information processing systems*, *33*, 1877–1901.

Burgess, J., Wang, K.-C., & Yeung, S. (2023). Viewpoint textual inversion: Unleashing novel view synthesis with pretrained 2d diffusion models. *arXiv preprint arXiv:2309.07986*.

Cao, M., Wang, X., Qi, Z., Shan, Y., Qie, X., & Zheng, Y. (2023). Masactrl: Tuning-free mutual self-attention control for consistent image synthesis and editing. *arXiv preprint arXiv:2304.08465*.

Cao, T., Kreis, K., Fidler, S., Sharp, N., & Yin, K. (2023). Texfusion: Synthesizing 3d textures with text-guided image diffusion models. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 4169–4181).

Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., & Joulin, A. (2021). Emerging properties in self-supervised vision transformers. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 9650–9660).

Chan, C., Durand, F., & Isola, P. (2022). Learning to generate line drawings that convey geometry and semantics. In *Cvpr*.

Chang, A. X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., ... others (2015). ShapeNet: An information-rich 3D model repository. *arXiv preprint arXiv:1512.03012*.

Chen, D. Z., Siddiqui, Y., Lee, H.-Y., Tulyakov, S., & Nießner, M. (2023). Text2tex: Text-driven texture synthesis via diffusion models. *arXiv preprint arXiv:2303.11396*.

Chen, M., Laina, I., & Vedaldi, A. (2023). Training-free layout control with cross-attention guidance. *arXiv preprint arXiv:2304.03373*.

Chen, W., Hu, H., Li, Y., Ruiz, N., Jia, X., Chang, M.-W., & Cohen, W. W. (2024). Subject-driven text-to-image generation via apprenticeship learning. *Advances in Neural Information Processing Systems*, *36*.

Chen, W., Ling, H., Gao, J., Smith, E., Lehtinen, J., Jacobson, A., & Fidler, S. (2019). Learning to predict 3d objects with an interpolation-based differentiable renderer. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, & R. Garnett (Eds.), *Advances in neural information processing systems.* Curran Associates, Inc.

Chen, Z., & Zhang, H. (2019). Learning implicit fields for generative shape modeling. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR).*

Cheng, T.-Y., Gadelha, M., Groueix, T., Fisher, M., Mech, R., Markham, A., & Trigoni, N. (2024). Learning continuous 3d words for text-to-image generation. *arXiv preprint arXiv:2402.08654*.

Cheng, T.-Y., Yang, H.-R., Trigoni, N., Chen, H.-T., & Liu, T.-L. (2022). Pose adaptive dual mixup for few-shot single-view 3d reconstruction. In *Proceedings of the aaai conference on artificial intelligence* (Vol. 36, pp. 427–435).

Choy, C. B., Xu, D., Gwak, J., Chen, K., & Savarese, S. (2016). 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In B. Leibe,

J. Matas, N. Sebe, & M. Welling (Eds.), *Eccv*.

Corneanu, C., Gadde, R., & Martinez, A. M. (2024). Latentpaint: Image inpainting in latent space with diffusion models. In *Proceedings of the ieee/cvf winter conference on applications of computer vision* (pp. 4334–4343).

Deitke, M., Liu, R., Wallingford, M., Ngo, H., Michel, O., Kusupati, A., ... others (2024). Objaverse-xl: A universe of 10m+ 3d objects. *Advances in Neural Information Processing Systems*, *36*.

Deitke, M., Schwenk, D., Salvador, J., Weihs, L., Michel, O., VanderBilt, E., ... Farhadi, A. (2023). Objaverse: A universe of annotated 3d objects. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 13142–13153).

Delanoy, J., Lagunas, M., Condor, J., Gutierrez, D., & Masia, B. (2022). A generative framework for image-based editing of material appearance using perceptual attributes. In *Computer graphics forum* (Vol. 41, pp. 453–464).

Deschaintre, V., Aittala, M., Durand, F., Drettakis, G., & Bousseau, A. (2019). Flexible svbrdf capture with a multi-image deep network. In *Computer graphics forum* (Vol. 38, pp. 1–13).

DeTone, D., Malisiewicz, T., & Rabinovich, A. (2018). Superpoint: Self-supervised interest point detection and description. In *Cvprw*.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Dhariwal, P., & Nichol, A. (2021). Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, *34*, 8780–8794.

Duggal, S., & Pathak, D. (2022). Topologically-aware deformation fields for single-view 3d reconstruction. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 1536–1546).

Dwibedi, D., Aytar, Y., Tompson, J., Sermanet, P., & Zisserman, A. (2021). With a little help from my friends: Nearest-neighbor contrastive learning of visual representations. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 9588–9597).

Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., Saini, H., ... others (2024). Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning.*

Fan, H., Su, H., & Guibas, L. J. (2017). A point set generation network for 3d object reconstruction from a single image. In *Cvpr.*

Fu, S., Tamir, N., Sundaram, S., Chai, L., Zhang, R., Dekel, T., & Isola, P. (2024). Dreamsim: Learning new dimensions of human visual similarity using synthetic data. *Advances in Neural Information Processing Systems*, *36*.

Gadelha, M., Maji, S., & Wang, R. (2017). 3d shape induction from 2d views of multiple objects. In *2017 international conference on 3d vision (3dv)* (pp. 402–411).

Gal, R., Alaluf, Y., Atzmon, Y., Patashnik, O., Bermano, A. H., Chechik, G., & Cohen-or, D. (2022). An image is worth one word: Personalizing text-to-image generation using textual inversion. In *The eleventh international conference on learning representations.*

Ganapathi-Subramanian, V., Diamanti, O., Pirk, S., Tang, C., Niessner, M., & Guibas, L. J. (2018, sep). Parsing geometry using structure-aware shape templates. In *2018 international conference on 3d vision (3dv)* (p. 672-681). Los Alamitos, CA, USA: IEEE Computer Society. Retrieved from https://doi.ieeecomputersociety.org/10.1109/3DV.2018.00082 doi: 10.1109/3DV.2018.00082

Gao*, R., Holynski*, A., Henzler, P., Brussee, A., Martin-Brualla, R., Srinivasan, P. P., ... Poole*, B. (2024). Cat3d: Create anything in 3d with multi-view

diffusion models. *arXiv*.

Ge, S., Park, T., Zhu, J.-Y., & Huang, J.-B. (2023). Expressive text-to-image generation with rich text. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 7545–7556).

Gkioxari, G., Malik, J., & Johnson, J. (2019). Mesh r-cnn. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 9785–9795).

Goel, S., Kanazawa, A., , & Malik, J. (2020). Shape and viewpoints without keypoints. In *Eccv*.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... Bengio, Y. (2020). Generative adversarial networks. *Communications of the ACM*, *63*(11), 139–144.

Grill, J.-B., Strub, F., Altché, F., Tallec, C., Richemond, P., Buchatskaya, E., ... others (2020). Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, *33*, 21271–21284.

Groueix, T., Fisher, M., Kim, V. G., Russell, B., & Aubry, M. (2018). AtlasNet: A Papier-Mâché Approach to Learning 3D Surface Generation. In *Proceedings ieee conf. on computer vision and pattern recognition (cvpr)*.

Guo, Y., Yang, C., Rao, A., Liang, Z., Wang, Y., Qiao, Y., ... Dai, B. (2023). Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. *arXiv preprint arXiv:2307.04725*.

Han, L., Li, Y., Zhang, H., Milanfar, P., Metaxas, D., & Yang, F. (2023). Svdiff: Compact parameter space for diffusion fine-tuning. *ICCV*.

He, K., Chen, X., Xie, S., Li, Y., Dollár, P., & Girshick, R. (2022). Masked autoencoders are scalable vision learners. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 16000–16009).

He, K., Fan, H., Wu, Y., Xie, S., & Girshick, R. (2020). Momentum contrast for

unsupervised visual representation learning. In *Cvpr* (pp. 9729–9738).

He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask r-cnn. In *Proceedings of the ieee international conference on computer vision* (pp. 2961–2969).

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Cvpr.*

Henderson, P., Tsiminaki, V., & Lampert, C. (2020). Leveraging 2D data to learn textured 3D mesh generation. In *Ieee conference on computer vision and pattern recognition (cvpr).*

Henzler, P., Mitra, N. J., & Ritschel, T. (2019, October). Escaping plato's cave: 3d shape from adversarial rendering. In *The ieee international conference on computer vision (iccv).*

Hertz, A., Mokady, R., Tenenbaum, J., Aberman, K., Pritch, Y., & Cohen-Or, D. (2022). Prompt-to-prompt image editing with cross attention control. In *arxiv preprint arxiv:2208.01626.*

Ho, J., Jain, A., & Abbeel, P. (2020). Denoising diffusion probabilistic models. *Advances in neural information processing systems*, *33*, 6840–6851.

Ho, J., Saharia, C., Chan, W., Fleet, D. J., Norouzi, M., & Salimans, T. (2022). Cascaded diffusion models for high fidelity image generation. *The Journal of Machine Learning Research*, *23*(1), 2249–2281.

Ho, J., & Salimans, T. (2022). Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*.

Hu, E. J., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., ... others (2021). Lora: Low-rank adaptation of large language models. In *International conference on learning representations.*

Hu, T., Wang, L., Xu, X., Liu, S., & Jia, J. (2021). Self-supervised 3d mesh reconstruction from single images. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 6002–6011).

Insafutdinov, E., & Dosovitskiy, A. (2018). Unsupervised learning of shape and pose with differentiable point clouds. In *Neurips*.

Jang, W., & Agapito, L. (2021). Codenerf: Disentangled neural radiance fields for object categories. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 12949–12958).

Jiang, Y., Yu, C., Cao, C., Wang, F., Hu, W., & Gao, J. (2024). Animate3d: Animating any 3d model with multi-view video diffusion. *arXiv preprint arXiv:2407.11398*.

Jimenez Rezende, D., Eslami, S. M. A., Mohamed, S., Battaglia, P., Jaderberg, M., & Heess, N. (2016). Unsupervised learning of 3d structure from images. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, & R. Garnett (Eds.), *Advances in neural information processing systems* (Vol. 29). Curran Associates, Inc. Retrieved from https://proceedings.neurips.cc/paper/2016/file/1d94108e907bb8311d8802b48fd54b4a-Paper.pdf

Kanazawa, A., Tulsiani, S., Efros, A. A., & Malik, J. (2018). Learning category-specific mesh reconstruction from image collections. In *Proceedings of the european conference on computer vision (eccv)* (pp. 371–386).

Kang, M., Zhu, J.-Y., Zhang, R., Park, J., Shechtman, E., Paris, S., & Park, T. (2023). Scaling up gans for text-to-image synthesis. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 10124–10134).

Kar, A., Tulsiani, S., Carreira, J., & Malik, J. (2015). Category-specific object reconstruction from a single image. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 1966–1974).

Karras, T., Aittala, M., Aila, T., & Laine, S. (2022). Elucidating the design space of diffusion-based generative models. *Advances in Neural Information*

Processing Systems, *35*, 26565–26577.

Kato, H., & Harada, T. (2019). Learning view priors for single-view 3d reconstruction. In *The ieee conference on computer vision and pattern recognition (cvpr).*

Kato, H., Ushiku, Y., & Harada, T. (2018). Neural 3d mesh renderer. In *The ieee conference on computer vision and pattern recognition (cvpr).*

Khan, E. A., Reinhard, E., Fleming, R. W., & Bülthoff, H. H. (2006). Image-based material editing. *ACM Transactions on Graphics (TOG)*, *25*(3), 654–663.

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., ... others (2023). Segment anything. *arXiv preprint arXiv:2304.02643*.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Neurips* (Vol. 25, pp. 1097–1105).

Kumari, N., Zhang, B., Zhang, R., Shechtman, E., & Zhu, J.-Y. (2023). Multi-concept customization of text-to-image diffusion. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 1931–1941).

Kuo, W., Angelova, A., Lin, T.-Y., & Dai, A. (2020). Mask2cad: 3d shape prediction by learning to segment and retrieve. In *European conference on computer vision* (pp. 260–277).

Kutulakos, K., & Seitz, S. (1999). A theory of shape by space carving. In *Proceedings of the seventh ieee international conference on computer vision.*

Li, B., Zheng, C., Zhu, W., Mai, J., Zhang, B., Wonka, P., & Ghanem, B. (2024). Vivid-zoo: Multi-view video generation with diffusion model. *arXiv preprint arXiv:2406.08659*.

Li, X., Liu, S., Kim, K., De Mello, S., Jampani, V., Yang, M.-H., & Kautz, J. (2020). Self-supervised single-view 3d reconstruction via semantic consistency. In *Eccv.*

Li, Y., Liu, H., Wu, Q., Mu, F., Yang, J., Gao, J., ... Lee, Y. J. (2023). Gligen: Open-set grounded text-to-image generation. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 22511–22521).

Li, Z., & Snavely, N. (2018). Learning intrinsic image decomposition from watching the world. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 9039–9048).

Liang, Y., Wakaki, R., Nobuhara, S., & Nishino, K. (2022). Multimodal material segmentation. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 19800–19808).

Lin, C.-H., Ma, W.-C., Torralba, A., & Lucey, S. (2021). Barf: Bundle-adjusting neural radiance fields. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 5741–5751).

Lin, C.-H., Wang, C., & Lucey, S. (2020). Sdf-srn: Learning signed distance 3d object reconstruction from static images. In *Advances in neural information processing systems (NeurIPS).*

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., ... Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *Computer vision–eccv 2014: 13th european conference, zurich, switzerland, september 6-12, 2014, proceedings, part v 13* (pp. 740–755).

Liu, M., Xu, C., Jin, H., Chen, L., Varma T, M., Xu, Z., & Su, H. (2024). One-2-3-45: Any single image to 3d mesh in 45 seconds without per-shape optimization. *Advances in Neural Information Processing Systems*, *36*.

Liu, R., Wu, R., Hoorick, B. V., Tokmakov, P., Zakharov, S., & Vondrick, C. (2023). *Zero-1-to-3: Zero-shot one image to 3d object.*

Liu, S., Li, T., Chen, W., & Li, H. (2019, Oct). Soft rasterizer: A differentiable renderer for image-based 3d reasoning. *The IEEE International Conference on Computer Vision (ICCV).*

Liu, Z., Zhang, Y., Shen, Y., Zheng, K., Zhu, K., Feng, R., ... Cao, Y. (2023). Cones 2: Customizable image synthesis with multiple subjects. *arXiv preprint arXiv:2305.19327*.

Longuet-Higgins, H. C. (1987). A computer algorithm for reconstructing a scene from two projections. In *Readings in computer vision: Issues, problems, principles, and paradigms.* Morgan Kaufmann Publishers Inc.

Lopes, I., Pizzati, F., & de Charette, R. (2023). Material palette: Extraction of materials from a single image. *arXiv preprint arXiv:2311.17060*.

LoweDavid, G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*.

Mandikal, P., Navaneet, K. L., Agarwal, M., & Babu, R. V. (2018). 3D-LMNet: Latent embedding matching for accurate and diverse 3d point cloud reconstruction from a single image. In *Proceedings of the british machine vision conference (BMVC).*

Marques, M., & Costeira, J. (2009, 02). Estimating 3d shape from degenerate sequences with missing data. *Computer Vision and Image Understanding*, *113*, 261-272. doi: 10.1016/j.cviu.2008.09.004

Martin-Brualla, R., Radwan, N., Sajjadi, M. S. M., Barron, J. T., Dosovitskiy, A., & Duckworth, D. (2021). NeRF in the Wild: Neural Radiance Fields for Unconstrained Photo Collections. In *Cvpr.*

Meka, A., Maximov, M., Zollhoefer, M., Chatterjee, A., Seidel, H.-P., Richardt, C., & Theobalt, C. (2018). Lime: Live intrinsic material estimation. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 6315–6324).

Mescheder, L., Oechsle, M., Niemeyer, M., Nowozin, S., & Geiger, A. (2019). Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the ieee/cvf conference on computer vision and pattern*

recognition (pp. 4460–4470).

Michalkiewicz, M., Parisot, S., Tsogkas, S., Baktashmotlagh, M., Eriksson, A., & Belilovsky, E. (2020). Few-shot single-view 3-d object reconstruction with compositional priors. In *European conference on computer vision* (pp. 614–630).

Michel, O., Bhattad, A., VanderBilt, E., Krishna, R., Kembhavi, A., & Gupta, T. (2024). Object 3dit: Language-guided 3d-aware image editing. *Advances in Neural Information Processing Systems*, *36*.

Mildenhall, B., Srinivasan, P. P., Tancik, M., Barron, J. T., Ramamoorthi, R., & Ng, R. (2020). Nerf: Representing scenes as neural radiance fields for view synthesis. In *European conference on computer vision* (pp. 405–421).

Mokady, R., Hertz, A., Aberman, K., Pritch, Y., & Cohen-Or, D. (2022). Null-text inversion for editing real images using guided diffusion models. *arXiv preprint arXiv:2211.09794*.

Monnier, T., Fisher, M., Efros, A. A., & Aubry, M. (2022). Share with thy neighbors: Single-view reconstruction by cross-instance consistency..

Mou, C., Wang, X., Xie, L., Zhang, J., Qi, Z., Shan, Y., & Qie, X. (2023). T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. *arXiv preprint arXiv:2302.08453*.

Müller, N., Simonelli, A., Porzi, L., Bulò, S. R., Nießner, M., & Kontschieder, 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* (pp. 3971–3980).

Myronenko, A., & Song, X. (2010). Point set registration: Coherent point drift. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *32*(12), 2262-2275. doi: 10.1109/TPAMI.2010.46

Niemeyer, M., Barron, J. T., Mildenhall, B., Sajjadi, M. S., Geiger, A., & 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* (pp. 5480–5490).

Niemeyer, M., Mescheder, L., Oechsle, M., & Geiger, A. (2020). Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In *Proc. ieee conf. on computer vision and pattern recognition (cvpr).*

Oechsle, M., Peng, S., & Geiger, A. (2021). Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. In *International conference on computer vision (iccv).*

Oord, A. v. d., Li, Y., & Vinyals, O. (2018). Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., ... others (2023). Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*.

Pandey, K., Guerrero, P., Gadelha, M., Hold-Geoffroy, Y., Singh, K., & Mitra, N. J. (2024). Diffusion handles: Enabling 3d edits for diffusion models by lifting activations to 3d. *CVPR*.

Park, J. J., Florence, P., Straub, J., Newcombe, R., & Lovegrove, S. (2019, June). Deepsdf: Learning continuous signed distance functions for shape representation. In *The ieee conference on computer vision and pattern recognition (cvpr).*

Parmar, G., Kumar Singh, K., Zhang, R., Li, Y., Lu, J., & Zhu, J.-Y. (2023). Zero-shot image-to-image translation. In *Acm siggraph 2023 conference proceedings* (pp. 1–11).

Pavllo, D., Spinks, G., Hofmann, T., Moens, M.-F., & Lucchi, A. (2020). Convolutional generation of textured 3d meshes. In *Neural information processing*

systems (neurips).

Podell, D., English, Z., Lacey, K., Blattmann, A., Dockhorn, T., Müller, J., ... Rombach, R. (2023). Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*.

Qin, X., Dai, H., Hu, X., Fan, D.-P., Shao, L., & Gool, L. V. (2022). Highly accurate dichotomous image segmentation. In *Eccv*.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., ... others (2021). Learning transferable visual models from natural language supervision. In *International conference on machine learning* (pp. 8748–8763).

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., ... Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, *21*(1), 5485–5551.

Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., & Chen, M. (2022). Hierarchical text-conditional image generation with clip latents. In *arxiv preprint arxiv:2204.06125* (Vol. 1, p. 3).

Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., ... Sutskever, I. (2021). Zero-shot text-to-image generation. In *International conference on machine learning* (pp. 8821–8831).

Ranftl, R., Bochkovskiy, A., & Koltun, V. (2021). Vision transformers for dense prediction. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 12179–12188).

Ravi, N., Gabeur, V., Hu, Y.-T., Hu, R., Ryali, C., Ma, T., ... Feichtenhofer, C. (2024). Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*. Retrieved from https://arxiv.org/abs/2408.00714

Real, E., Shlens, J., Mazzocchi, S., Pan, X., & Vanhoucke, V. (2017). Youtube-

boundingboxes: A large high-precision human-annotated data set for object detection in video. In *proceedings of the ieee conference on computer vision and pattern recognition* (pp. 5296–5305).

Reizenstein, J., Shapovalov, R., Henzler, P., Sbordone, L., Labatut, P., & Novotny, D. (2021). Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction. In *International conference on computer vision.*

Ren, S., He, K., Girshick, R., & Sun, J. (2016). Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE transactions on pattern analysis and machine intelligence*, *39*(6), 1137–1149.

Richardson, E., Metzer, G., Alaluf, Y., Giryes, R., & Cohen-Or, D. (2023). Texture: Text-guided texturing of 3d shapes. *arXiv preprint arXiv:2302.01721*.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., & Ommer, B. (2021). *High-resolution image synthesis with latent diffusion models.*

Ruiz, N., Li, Y., Jampani, V., Pritch, Y., Rubinstein, M., & Aberman, K. (2023). Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 22500–22510).

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., ... Fei-Fei, L. (2015). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, *115*(3), 211-252. doi: 10.1007/s11263-015-0816-y

Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E. L., ... others (2022). Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, *35*, 36479–36494.

Sarlin, P.-E., DeTone, D., Malisiewicz, T., & Rabinovich, A. (2020). SuperGlue:

Learning feature matching with graph neural networks. In *Cvpr.* Retrieved from https://arxiv.org/abs/1911.11763

Schuhmann, C., Beaumont, R., Vencu, R., Gordon, C., Wightman, R., Cherti, M., ... others (2022). Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems*, *35*, 25278–25294.

Sharma, P., Jampani, V., Li, Y., Jia, X., Lagun, D., Durand, F., ... Matthews, M. (2023). Alchemist: Parametric control of material properties with diffusion models. *arXiv preprint arXiv:2312.02970*.

Sharma, P., Philip, J., Gharbi, M., Freeman, B., Durand, F., & Deschaintre, V. (2023). Materialistic: Selecting similar materials in images. *ACM Transactions on Graphics (TOG)*, *42*(4), 1–14.

Shi, R., Chen, H., Zhang, Z., Liu, M., Xu, C., Wei, X., ... Su, H. (2023). *Zero123++: a single image to consistent multi-view diffusion base model.*

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Snavely, N., Seitz, S. M., & Szeliski, R. (2006). Photo tourism: exploring photo collections in 3d. In *Acm siggraph 2006 papers* (pp. 835–846).

Song, J., Meng, C., & Ermon, S. (2020). Denoising diffusion implicit models. In *International conference on learning representations.*

Song, Y., & Ermon, S. (2019). Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, *32*.

Subias, J. D., & Lagunas, M. (2023). In-the-wild material appearance editing using perceptual attributes. In *Computer graphics forum* (Vol. 42, pp. 333–345).

Sun, X., Wu, J., Zhang, X., Zhang, Z., Zhang, C., Xue, T., ... Freeman, W. T. (2018). Pix3d: Dataset and methods for single-image 3d shape modeling. In *Proceedings of the ieee conference on computer vision and pattern*

recognition (pp. 2974–2983).

Tomasi, C., & Kanade, T. (1992). Shape and motion from image streams under orthography: A factorization method. *IJCV*, *9*.

Triggs, B. (1996). Factorization methods for projective structure and motion. In *Cvpr.*

Tsai, Y.-H., Yang, M.-H., & Black, M. J. (2016). Video segmentation via object flow. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 3899–3908).

Tulsiani, S., Efros, A. A., & Malik, J. (2018). Multi-view consistency as supervisory signal for learning shape and pose prediction. In *Computer vision and pattern regognition (cvpr).*

Tulsiani, S., Kulkarni, N., & Gupta, A. (2020). Implicit mesh reconstruction from unannotated image collections. In *arxiv.*

Tulsiani, S., Zhou, T., Efros, A. A., & Malik, J. (2017). Multi-view supervision for single-view reconstruction via differentiable ray consistency. In *Computer vision and pattern regognition (cvpr).*

Upchurch, P., & Niu, R. (2022). A dense material segmentation dataset for indoor and outdoor scene parsing. In *European conference on computer vision* (pp. 450–466).

Uy, M. A., Pham, Q.-H., Hua, B.-S., Nguyen, T., & Yeung, S.-K. (2019). Revisiting point cloud classification: A new benchmark dataset and classification model on real-world data. In *Iccv* (pp. 1588–1597).

Van Hoorick, B., Wu, R., Ozguroglu, E., Sargent, K., Liu, R., Tokmakov, P., ... Vondrick, C. (2024). Generative camera dolly: Extreme monocular dynamic novel view synthesis. *arXiv preprint arXiv:2405.14868*.

Vicente, S., Carreira, J., Agapito, L., & Batista, J. (2014). Reconstructing pascal voc. In *2014 ieee conference on computer vision and pattern recognition.*

Voynov, A., Chu, Q., Cohen-Or, D., & Aberman, K. (2023). $p+$: Extended textual conditioning in text-to-image generation. *arXiv preprint arXiv:2303.09522*.

Wah, C., Branson, S., Welinder, P., Perona, P., & Belongie, S. (2011). The caltech-ucsd birds-200-2011 dataset.

Wallace, B., & Hariharan, B. (2019). Few-shot generalization for single-image 3d reconstruction via priors. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 3818–3827).

Wang, B., Ma, L., Zhang, W., & Liu, W. (2018). Reconstruction network for video captioning. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 7622–7631).

Wang, N., Zhang, Y., Li, Z., Fu, Y., Liu, W., & Jiang, Y.-G. (2018). Pixel2mesh: Generating 3d mesh models from single rgb images. In *Eccv*.

Wang, P., Liu, L., Liu, Y., Theobalt, C., Komura, T., & Wang, W. (2021). Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *NeurIPS*.

Wang, T. Y., Ritschel, T., & Mitra, N. J. (2018). Joint material and illumination estimation from photo sets in the wild. In *2018 international conference on 3d vision (3dv)* (pp. 22–31).

Wang, X., Darrell, T., Rambhatla, S. S., Girdhar, R., & Misra, I. (2024). Instancediffusion: Instance-level control for image generation. *arXiv preprint arXiv:2402.03290*.

Wang, Z., Wu, S., Xie, W., Chen, M., & Prisacariu, V. A. (2021). Nerf–: Neural radiance fields without known camera parameters. *arXiv preprint arXiv:2102.07064*.

Wu, R., Mildenhall, B., Henzler, P., Park, K., Gao, R., Watson, D., ... Holynski, A. (2023). Reconfusion: 3d reconstruction with diffusion priors. *arXiv*.

Wu, S., Makadia, A., Wu, J., Snavely, N., Tucker, R., & Kanazawa, A. (2021).

De-rendering the world's revolutionary artefacts. In *Cvpr*.

Wu, T., Zhang, J., Fu, X., Wang, Y., Ren, J., Pan, L., ... others (2023). 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* (pp. 803–814).

Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., & Xiao, J. (2015). 3d shapenets: A deep representation for volumetric shapes. In *Cvpr*.

Xie, H., Yao, H., Sun, X., Zhou, S., & 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* (pp. 2690–2698).

Xie, H., Yao, H., Zhang, S., Zhou, S., & 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.

Xie, Y., Yao, C.-H., Voleti, V., Jiang, H., & Jampani, V. (2024). Sv4d: Dynamic 3d content generation with multi-frame and multi-view consistency. *arXiv preprint arXiv:2407.17470*.

Xu, Y.-S., Fu, T.-J., Yang, H.-K., & Lee, C.-Y. (2018). Dynamic video segmentation network. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 6556–6565).

Yan, X., Yang, J., Yumer, E., Guo, Y., & Lee, H. (2016). Perspective transformer nets: Learning single-view 3d object reconstruction without 3d supervision. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, & R. Garnett (Eds.), *Advances in neural information processing systems 29* (pp. 1696–1704). Curran Associates, Inc. Retrieved from http://papers.nips.cc/paper/6206-perspective-transformer-nets-learning-single-view-3d-object-reconstruction-without-3d-supervision.pdf

Yang, S., Xu, M., Xie, H., Perry, S., & Xia, J. (2021). Single-view 3d object reconstruction from shape priors in memory. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 3152–3161).

Yang, Z., Wang, J., Gan, Z., Li, L., Lin, K., Wu, C., ... others (2023). Reco: Region-controlled text-to-image generation. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 14246–14255).

Yariv, L., Gu, J., Kasten, Y., & Lipman, Y. (2021). Volume rendering of neural implicit surfaces. In *Thirty-fifth conference on neural information processing systems.*

Yariv, L., Kasten, Y., Moran, D., Galun, M., Atzmon, M., Ronen, B., & Lipman, Y. (2020). Multiview neural surface reconstruction by disentangling geometry and appearance. *Advances in Neural Information Processing Systems*, *33*.

Ye, H., Zhang, J., Liu, S., Han, X., & Yang, W. (2023). Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. *arXiv preprint arXiv:2308.06721*.

Ye, Y., Tulsiani, S., & Gupta, A. (2021). Shelf-supervised mesh prediction in the wild. In *Computer vision and pattern recognition (cvpr).*

Yeh, C.-H., Hong, C.-Y., Hsu, Y.-C., Liu, T.-L., Chen, Y., & LeCun, Y. (2022). Decoupled contrastive learning..

Yeh, Y.-Y., Huang, J.-B., Kim, C., Xiao, L., Nguyen-Phuoc, T., Khan, N., ... others (2024). Texturedreamer: Image-guided texture synthesis through geometry-aware diffusion. *arXiv preprint arXiv:2401.09416*.

Yoo, P., Guo, J., Matsuo, Y., & Gu, S. S. (2023). Dreamsparse: Escaping from plato's cave with 2d diffusion model given sparse views. *arXiv preprint arXiv:2306.03414*.

Yu, A., Ye, V., Tancik, M., & Kanazawa, A. (2021). pixelnerf: Neural radiance fields from one or few images. In *Proceedings of the ieee/cvf conference on*

computer vision and pattern recognition (pp. 4578–4587).

Zeng, Z., Deschaintre, V., Georgiev, I., Hold-Geoffroy, Y., Hu, Y., Luan, F., ... Hašan, M. (2024). RgbâĘŤ x: Image decomposition and synthesis using material-and lighting-aware diffusion models. In *Acm siggraph 2024 conference papers* (pp. 1–11).

Zhang, J., Yao, Y., & Quan, L. (2021). Learning signed distance field for multi-view surface reconstruction. *International Conference on Computer Vision (ICCV)*.

Zhang, L., Rao, A., & Agrawala, M. (2023). Adding conditional control to text-to-image diffusion models. In *Ieee international conference on computer vision (iccv)*.

Zhang, R., Isola, P., Efros, A. A., Shechtman, E., & Wang, O. (2018). The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 586–595).

Zhao, S., Chen, D., Chen, Y.-C., Bao, J., Hao, S., Yuan, L., & Wong, K.-Y. K. (2024). Uni-controlnet: All-in-one control to text-to-image diffusion models. *Advances in Neural Information Processing Systems*, *36*.

Zhou, Y., Barnes, C., Lu, J., Yang, J., & Li, H. (2019). On the continuity of rotation representations in neural networks. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 5745–5753).