# 3DMiner: Discovering Shapes from Large-Scale Unannotated Image Datasets

Ta-Ying Cheng[1*] Matheus Gadelha[2] Sören Pirk[2] Thibault Groueix[2]
Radomír Měch[2] Andrew Markham[1] Niki Trigoni[1]

[1]University of Oxford [2]Adobe Research

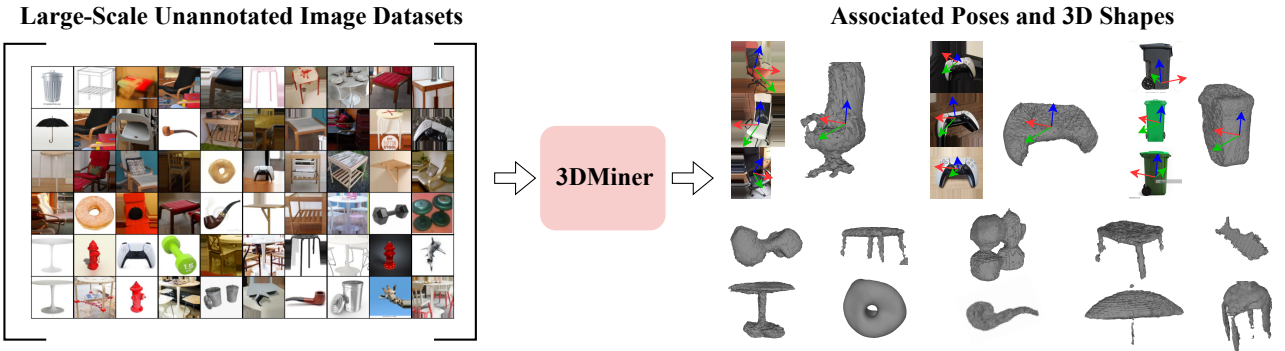


Figure 1: **Overview**. We present 3DMiner, a scalable framework designed to obtain associating poses and reconstruct shapes from *diverse and realistic* sets of images *without any 3D data, pose annotation, camera information, or keypoints*.

## Abstract

*We present 3DMiner – a pipeline for mining 3D shapes from challenging large-scale unannotated image datasets. 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.*

## 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

[*]Work was partially done during internship at Adobe Research.

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 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 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.

## 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 [31], SfM has been extensively researched [51, 52, 49, 1]. Before optimization, the camera parameters are typically initialized by either applying RANSAC [3] 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 [32] have been outperformed by learned approaches [10, 47]. Interestingly, features from self-supervised approaches, like DINO-ViT [4], have proven to be robust in a wide range of downstream tasks, including keypoint matching [2].

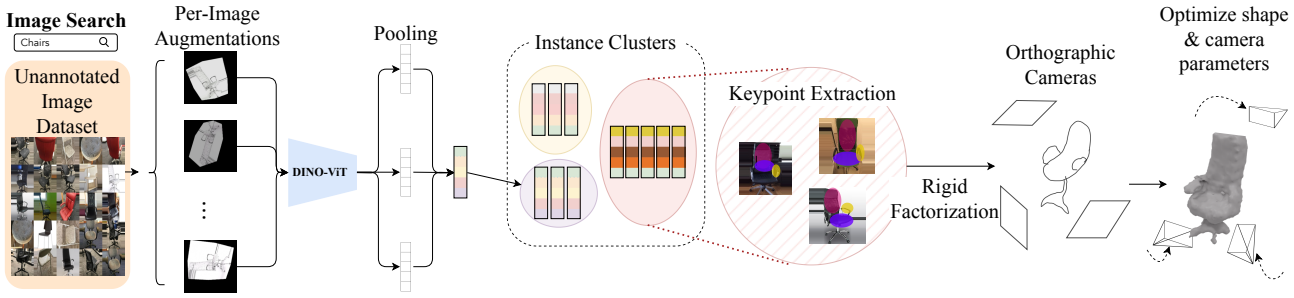


Figure 2: **3DMiner pipeline.** Our method starts by grouping images that depict similar 3D shapes, regardless of the texture of the shape, the camera view-point or the background. To do so, we perturb each image with various transformations (e.g. color jittering, perspective and rotation) and we pool 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.

More recently, Mildenhall *et al.* [37] demonstrated photo-realistic performance on the task of novel-view synthesis using differentiable volume rendering of neural radiance fields. NeRFs [37] initially assumed multiple views of the same stationary scene with known camera poses and several extensions are relevant to our work. NeRF-W [35] 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 [68, 65, 59, 64] or occupancy fields [41, 42]. BARF [28] assumes coarse camera poses, and jointly performs bundle-adjustment and optimizes the radiance field. Like BARF, we jointly optimize an occupancy field and perform a bundle-adjustment.

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 [4] features to estimate corresponding keypoints across images with different textures and backgrounds, and use those keypoints to estimate coarse camera poses. Similar to BARF [28] and space carving [26], 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 [5], ModelNet [61], Pix3D [50], to learn 3D reconstructions. Their most distinctive feature is usually the choice of 3D representation. 3D geometry can be represented via voxel grids [9, 62, 8], point clouds [12, 33], meshes [16, 58, 14], or implicit functions [36, 7, 43, 67]. 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-Image 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 [55, 63, 21, 41] or differentiable mesh renderers [25, 30, 6]. Notably, [53, 20] 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 [23, 56, 55, 22, 11, 29, 17], silhouette losses [22, 15, 54, 19, 60, 66], shape templates [15, 54], different forms of symmetry [17, 15, 22, 27, 19] including rotation symmetries [60]. Several approaches leverage off-the-shelf 2D networks [27, 66] or generative adversarial techniques [18, 13, 24, 44, 66]. Particularly relevant to us are Unicorn [38] and SMR [19]. In SMR, Hu *et al.* [19] use self-supervised learning to learn 3D from 2D images. Their method requires only images and their corresponding silhouettes. In Unicorn, Monnier *et al.* [38] 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 challeng-

ing datasets. Most comparisons mainly exist on synthetic data [5] or very constrained real-world images [57], where the background and foreground are very distinct and not many different shapes can be found. Previous approaches cannot handle data such as Pix3D [50] (unless trained with additional datasets), not to mention in-the-wild datasets such as LAION-5B [48]. 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.

# 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 2 shows an overview of our method. In the following subsections we describe each step of this pipeline in detail.

## 3.1. Clustering similar shapes

We aim to find clusters of similar shapes, irrespective of their texture, background and illumination conditions. Caron *et al.* [4] 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 [46] 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:

$$z_I = [\max(S_I), \text{mean}(S_I), \min(S_I)]. \quad (1)$$

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 1, we perform a bottom-up agglomerative clustering. Section 4.4 presents an analysis of our augmentation scheme on the Pix3D datase and Figure 5 shows examples of clusters from the LAION-5B dataset.

## 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 [34]. 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 [22, 56, 23]. 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 [2]. 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 a multiple clusters.

Equipped with estimated keypoints within a cluster, we then perform rigid factorisation through SVD and Stiefel manifold projections following [34] to obtain an orthographic camera for every image $I$ in the format:

$$p_{2d} = M \cdot p_{3d} + t, \quad (2)$$

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 $I$.

| | | | Threshold | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | 0.5 | | 0.4 | | 0.3 | |
| | **CD** ↓ | **F1** ↑ | **CD** ↓ | **F1** ↑ | **CD** ↓ | **F1** ↑ | **CD** ↓ | **F1** ↑ |
| SMR [19] | 0.192 | 0.130 | 0.189 | 0.131 | 0.188 | 0.132 | 0.267 | 0.110 |
| Unicorn [38] | 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 1: **Comparisons on Pix3D Chairs.** We align the meshes to their ground truths via Coherent Point Drift [39] 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 (top), 3DMiner improves by 33% the CD and 10 points the F1, and the performance increases significantly when we use our selection criterion (see Section 4.2 for more discussion).

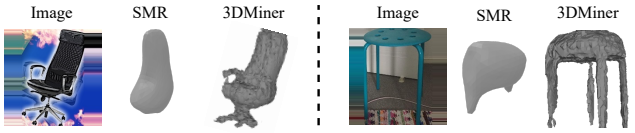


Figure 3: **Qualitative Comparison on Pix3D chairs.** When applied to actual in-the-wild images of objects, 3DMiner generates much more accurate geometry than state-of-art methods like SMR.

### 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 of NeRF [37] 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 [45] 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 [28] 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 2) to initialise a perspective camera-to-world matrix $P$ in $\mathbb{R}^{4\times4}$:

$$P = \begin{bmatrix} \frac{m_1}{||m_1||} & | \\ \frac{m_2-(p_1\cdot m_2)p_1}{||m_2-(p_1\cdot m_2)p_1||} & T \\ p_1 \times p_2 & | \\ 0 & 1 \end{bmatrix}^{-1}, \tag{3}$$

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 [69]. 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 guaranty 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 pixels, and sample points along each ray $r$. Each point $x$ is encoded with the progressive positional encoding technique from BARF [28] where the $k^{th}$ frequency of the positional encoding is:

$$\gamma(x, \alpha) = w(\alpha) \cdot [\cos 2^k \pi x, \sin 2^k \pi x], \tag{4}$$

where $w$ is a weight controlled by hyperparameter $\alpha$ 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:

$$\mathcal{L}_r = \text{BCE}(o_{gt}, 1 - e^{(-\sum_{x\in r} \text{MLP}(x, [\gamma(x,\alpha)]_{\alpha=0}^{10}))}). \tag{5}$$

We jointly optimize the 3D occupancy network, and the bundle-adjustment parameters $f$, $M$, and $T$s for each image. We do not directly optimize the matrix $P$ via gradient

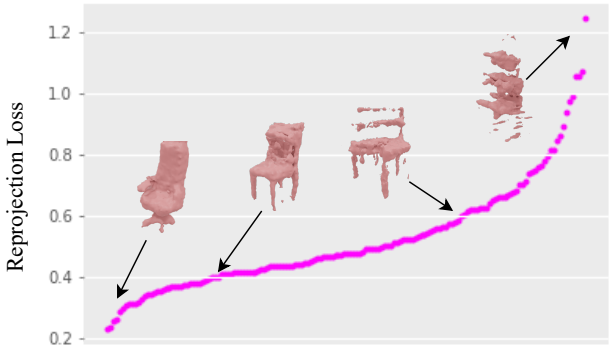


Figure 4: **Reprojection error.** We plot the reprojection error per cluster (averaged over each image in the cluster) in ascending order and show representative reconstructions for four data points. We empirically observe that the reprojection error is a good indicator of the quality of the reconstruction.

descent because it needs to remain in the manifold of rotation matrices. We follow [28] and use marching cube to extract a mesh from the learned occupancy field.

**Regularizing the 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 [40] and impose two extra regularizations during our occupancy field optimization. First, we encourage piece-wise smoothness of objects by imposing an additional geometric regularizer $\mathcal{L}_g$:

$$\mathcal{L}_g = (d(r_{i,j}) - d(r_{i,j+1}))^2 + (d(r_{i,j}) - d(r_{i+1,j}))^2, \quad (6)$$

where $r_{i,j}$ indicates the ray casted from pixel coordinate $(i, j)$ and $d$ is expected depth calculated in the same manner as [40]. 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.

# 4. Experiments

## 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 segments unless specified. We sample 32 rays and 32 points on each ray during every iteration and train the model for 300 epochs with $\alpha$ 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.

## 4.2. Comparison on Pix3D chairs

To validate our approach, we directly train the model on the very challenging Pix3D chairs dataset [50] 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 [11, 62] (please see Appendix for additional Pix3D categories). We compare against two state-of-the-art approaches: SMR [19] and Unicorn [38]. 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 [19] and Unicorn [38] 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 [39], 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 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, 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 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 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

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

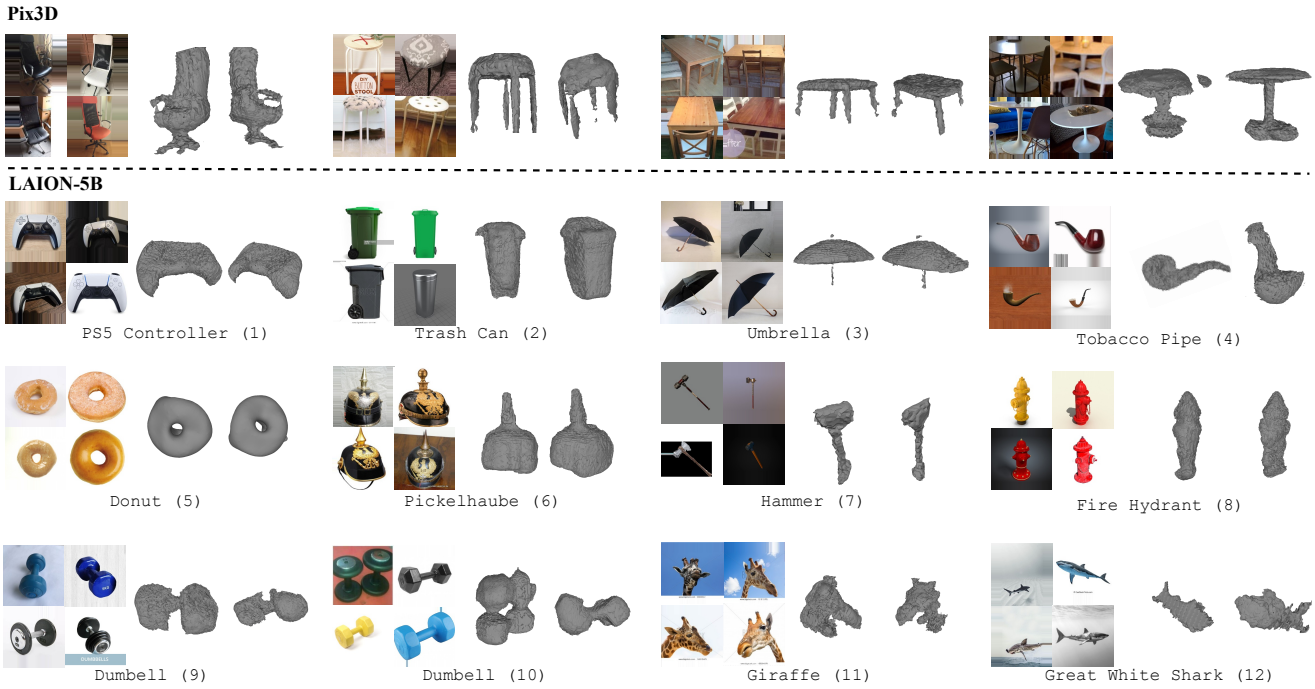


Figure 5: **Qualitative Results on Pix3D and LAION-5B.** We show examples of mining 3D shape from the in-the-wild LAION-5B dataset with the above text prompts. 1∼6 show reconstructions with high fidelity when clusters are very accurate. 7∼10 present generic shapes captured when the clusters are more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

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 1, we select images whose cluster has 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.

### 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 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 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 5, the image from LAION-5B are noisy, but our clustering leads to cleaner subsets of images. Specifically, reconstructions 1∼6 shows reconstructions with 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∼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

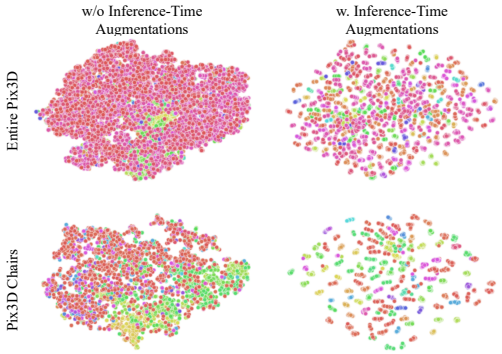


Figure 6: **Ablation of our augmentation scheme.** We show the T-SNE embeddings of Pix3D images embedded with DINO-ViT [4] with and without our augmentations scheme. Each color denotes a different object. Note that with augmentations, images corresponding to the same 3D object are grouped together, forming clusters on a single color (right), which is not the case without augmentations (left).

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

| 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 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 3.1), and "Dist. Thres." denotes the threshold distance set for agglomerative clustering. Our augmentation scheme improves the quality of the cluster (+11 points).

Figure 7: Two clusters from text prompt `Umbrella`, showing non-rigid objects separated based on part-wise poses.

both images are valid. Finally, we also investigated how the method would perform for non-rigid objects (`11`∼`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`.

### 4.4. Analysis

**Augmentations before image clustering.** We validate the contribution of our augmentation scheme on Pix3D images. Figure 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 the normalised mutual information (NMI) with the ground-truth clusters. Table 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.

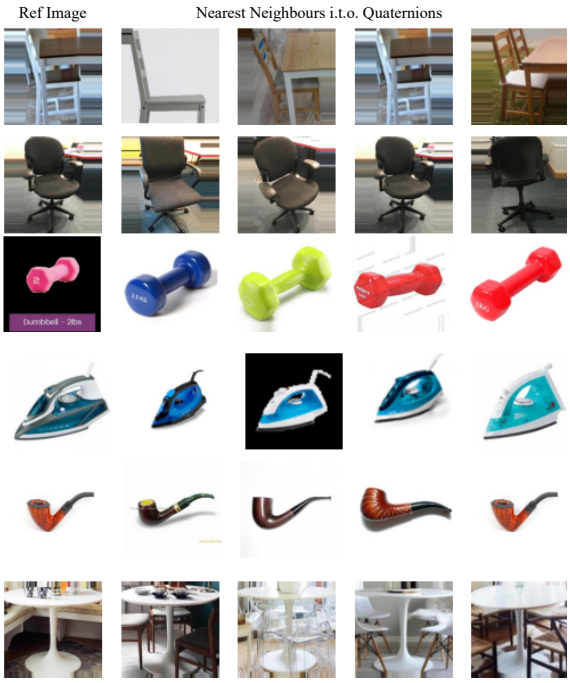


Figure 8: **Pose estimation Analysis.** We randomly select a reference image (leftmost column) out of a cluster and find the images with the nearest quaternion poses (rightmost 4 columns).

**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 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 5). Tightening the clustering threshold would also constrain rigidity within a cluster, though with the trade off 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).

**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 8. 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.

## 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 [56] 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.

## References

[1] Sameer Agarwal, Noah Snavely, Ian Simon, Steven M. Seitz, and Rick Szeliski. Building rome in a day. In *ICCV*, September 2009. 2

[2] Shir Amir, Yossi Gandelsman, Shai Bagon, and Tali Dekel. Deep vit features as dense visual descriptors. *arXiv preprint arXiv:2112.05814*, 2021. 2, 4

[3] Robert C. Bolles and Martin A. Fischler. 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*, IJCAI'81. Morgan Kaufmann Publishers Inc., 1981. 2

[4] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9650–9660, 2021. 2, 3, 4, 7

[5] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. ShapeNet: An information-rich 3D model repository. *arXiv preprint arXiv:1512.03012*, 2015. 3, 4

[6] Wenzheng Chen, Huan Ling, Jun Gao, Edward Smith, Jaakko Lehtinen, Alec Jacobson, and Sanja Fidler. Learning to predict 3d objects with an interpolation-based differentiable renderer. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2019. 3

[7] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 3

[8] Ta-Ying Cheng, Hsuan-Ru Yang, Niki Trigoni, Hwann-Tzong Chen, and Tyng-Luh Liu. Pose adaptive dual mixup for few-shot single-view 3d reconstruction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 427–435, 2022. 3

[9] Christopher B. Choy, Danfei Xu, JunYoung Gwak, Kevin Chen, and Silvio Savarese. 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *ECCV*, 2016. 3

[10] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *CVPRW*, 2018. 2

[11] Shivam Duggal and Deepak Pathak. Topologically-aware deformation fields for single-view 3d reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1536–1546, 2022. 3, 6

[12] Haoqiang Fan, Hao Su, and Leonidas J. Guibas. A point set generation network for 3d object reconstruction from a single image. In *CVPR*, 2017. 3

[13] Matheus Gadelha, Subhransu Maji, and Rui Wang. 3d shape induction from 2d views of multiple objects. In *2017 International Conference on 3D Vision (3DV)*, pages 402–411. IEEE, 2017. 3

[14] Vignesh Ganapathi-Subramanian, Olga Diamanti, Sören Pirk, Chengcheng Tang, Matthias Niessner, and Leonidas J. Guibas. Parsing geometry using structure-aware shape templates. In *2018 International Conference on 3D Vision (3DV)*, pages 672–681, Los Alamitos, CA, USA, sep 2018. IEEE Computer Society. 3

[15] Shubham Goel, Angjoo Kanazawa, , and Jitendra Malik. Shape and viewpoints without keypoints. In *ECCV*, 2020. 3

[16] Thibault Groueix, Matthew Fisher, Vladimir G. Kim, Bryan Russell, and Mathieu Aubry. AtlasNet: A Papier-Mâché Approach to Learning 3D Surface Generation. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2018. 3

[17] Paul Henderson, Vagia Tsiminaki, and Christoph Lampert. Leveraging 2D data to learn textured 3D mesh generation.

In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 3

[18] Philipp Henzler, Niloy J. Mitra, and Tobias Ritschel. Escaping plato's cave: 3d shape from adversarial rendering. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019. 3

[19] Tao Hu, Liwei Wang, Xiaogang Xu, Shu Liu, and Jiaya Jia. Self-supervised 3d mesh reconstruction from single images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6002–6011, 2021. 3, 5, 6

[20] Eldar Insafutdinov and Alexey Dosovitskiy. Unsupervised learning of shape and pose with differentiable point clouds. In *NeurIPS*, 2018. 3

[21] Danilo Jimenez Rezende, S. M. Ali Eslami, Shakir Mohamed, Peter Battaglia, Max Jaderberg, and Nicolas Heess. Unsupervised learning of 3d structure from images. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. 3

[22] Angjoo Kanazawa, Shubham Tulsiani, Alexei A. Efros, and Jitendra Malik. Learning category-specific mesh reconstruction from image collections. In *ECCV*, 2018. 3, 4

[23] Abhishek Kar, Shubham Tulsiani, Joao Carreira, and Jitendra Malik. Category-specific object reconstruction from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1966–1974, 2015. 3, 4

[24] Hiroharu Kato and Tatsuya Harada. Learning view priors for single-view 3d reconstruction. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 3

[25] Hiroharu Kato, Yoshitaka Ushiku, and Tatsuya Harada. Neural 3d mesh renderer. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 3

[26] K.N. Kutulakos and S.M. Seitz. A theory of shape by space carving. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, 1999. 3

[27] Xueting Li, Sifei Liu, Kihwan Kim, Shalini De Mello, Varun Jampani, Ming-Hsuan Yang, and Jan Kautz. Self-supervised single-view 3d reconstruction via semantic consistency. In *ECCV*, 2020. 3

[28] Chen-Hsuan Lin, Wei-Chiu Ma, Antonio Torralba, and Simon Lucey. Barf: Bundle-adjusting neural radiance fields. In *IEEE International Conference on Computer Vision (ICCV)*, 2021. 3, 5, 6

[29] Chen-Hsuan Lin, Chaoyang Wang, and Simon Lucey. Sdf-srn: Learning signed distance 3d object reconstruction from static images. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020. 3

[30] Shichen Liu, Tianye Li, Weikai Chen, and Hao Li. Soft rasterizer: A differentiable renderer for image-based 3d reasoning. *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2019. 3

[31] H. C. Longuet-Higgins. *A Computer Algorithm for Reconstructing a Scene from Two Projections*. Morgan Kaufmann Publishers Inc., 1987. 2

[32] G LoweDavid. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 2004. 2

[33] Priyanka Mandikal, K L Navaneet, Mayank Agarwal, and R Venkatesh Babu. 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)*, 2018. 3

[34] Manuel Marques and Joao Costeira. Estimating 3d shape from degenerate sequences with missing data. *Computer Vision and Image Understanding*, 113:261–272, 02 2009. 4

[35] Ricardo Martin-Brualla, Noha Radwan, Mehdi S. M. Sajjadi, Jonathan T. Barron, Alexey Dosovitskiy, and Daniel Duckworth. NeRF in the Wild: Neural Radiance Fields for Unconstrained Photo Collections. In *CVPR*, 2021. 3

[36] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4460–4470, 2019. 3

[37] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *European conference on computer vision*, pages 405–421. Springer, 2020. 3, 5

[38] Tom Monnier, Matthew Fisher, Alexei A Efros, and Mathieu Aubry. Share with thy neighbors: Single-view reconstruction by cross-instance consistency. 2022. 3, 5, 6

[39] Andriy Myronenko and Xubo Song. Point set registration: Coherent point drift. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(12):2262–2275, 2010. 5, 6

[40] Michael Niemeyer, Jonathan T Barron, Ben Mildenhall, Mehdi SM Sajjadi, Andreas Geiger, and Noha Radwan. Regnerf: Regularizing neural radiance fields for view synthesis from sparse inputs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5480–5490, 2022. 6

[41] Michael Niemeyer, Lars Mescheder, Michael Oechsle, and Andreas Geiger. Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020. 3

[42] Michael Oechsle, Songyou Peng, and Andreas Geiger. Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. In *International Conference on Computer Vision (ICCV)*, 2021. 3

[43] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 3

[44] Dario Pavllo, Graham Spinks, Thomas Hofmann, Marie-Francine Moens, and Aurelien Lucchi. Convolutional generation of textured 3d meshes. In *Neural Information Processing Systems (NeurIPS)*, 2020. 3

[45] Xuebin Qin, Hang Dai, Xiaobin Hu, Deng-Ping Fan, Ling Shao, and Luc Van Gool. Highly accurate dichotomous image segmentation. In *ECCV*, 2022. 5

[46] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. 4

[47] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. SuperGlue: Learning feature matching with graph neural networks. In *CVPR*, 2020. 2

[48] Christoph Schuhmann, Romain Beaumont, Cade W Gordon, Ross Wightman, Theo Coombes, Aarush Katta, Clayton Mullis, Patrick Schramowski, Srivatsa R Kundurthy, Katherine Crowson, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. 4

[49] Noah Snavely, Steven M Seitz, and Richard Szeliski. Photo tourism: exploring photo collections in 3d. In *ACM siggraph 2006 papers*, pages 835–846. 2006. 2

[50] Xingyuan Sun, Jiajun Wu, Xiuming Zhang, Zhoutong Zhang, Chengkai Zhang, Tianfan Xue, Joshua B Tenenbaum, and William T Freeman. Pix3d: Dataset and methods for single-image 3d shape modeling. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2974–2983, 2018. 3, 4, 6

[51] Carlo Tomasi and Takeo Kanade. Shape and motion from image streams under orthography: A factorization method. *IJCV*, 9, 1992. 2

[52] B. Triggs. Factorization methods for projective structure and motion. In *CVPR*, 1996. 2

[53] Shubham Tulsiani, Alexei A. Efros, and Jitendra Malik. Multi-view consistency as supervisory signal for learning shape and pose prediction. In *Computer Vision and Pattern Regognition (CVPR)*, 2018. 3

[54] Shubham Tulsiani, Nilesh Kulkarni, and Abhinav Gupta. Implicit mesh reconstruction from unannotated image collections. In *arXiv*, 2020. 3

[55] Shubham Tulsiani, Tinghui Zhou, Alexei A. Efros, and Jitendra Malik. Multi-view supervision for single-view reconstruction via differentiable ray consistency. In *Computer Vision and Pattern Regognition (CVPR)*, 2017. 3

[56] Sara Vicente, João Carreira, Lourdes Agapito, and Jorge Batista. Reconstructing pascal voc. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014. 3, 4, 9

[57] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011. 4

[58] Nanyang Wang, Yinda Zhang, Zhuwen Li, Yanwei Fu, Wei Liu, and Yu-Gang Jiang. Pixel2mesh: Generating 3d mesh models from single rgb images. In *ECCV*, 2018. 3

[59] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *NeurIPS*, 2021. 3

[60] Shangzhe Wu, Ameesh Makadia, Jiajun Wu, Noah Snavely, Richard Tucker, and Angjoo Kanazawa. De-rendering the world's revolutionary artefacts. In *CVPR*, 2021. 3

[61] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *CVPR*, 2015. 3

[62] Haozhe Xie, Hongxun Yao, Shengping Zhang, Shangchen Zhou, and Wenxiu Sun. Pix2vox++: Multi-scale context-aware 3d object reconstruction from single and multiple images. *International Journal of Computer Vision*, 128(12):2919–2935, 2020. 3, 6

[63] Xinchen Yan, Jimei Yang, Ersin Yumer, Yijie Guo, and Honglak Lee. Perspective transformer nets: Learning single-view 3d object reconstruction without 3d supervision. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 1696–1704. Curran Associates, Inc., 2016. 3

[64] Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. Volume rendering of neural implicit surfaces. In *Thirty-Fifth Conference on Neural Information Processing Systems*, 2021. 3

[65] Lior Yariv, Yoni Kasten, Dror Moran, Meirav Galun, Matan Atzmon, Basri Ronen, and Yaron Lipman. Multiview neural surface reconstruction by disentangling geometry and appearance. *Advances in Neural Information Processing Systems*, 33, 2020. 3

[66] Yufei Ye, Shubham Tulsiani, and Abhinav Gupta. Shelf-supervised mesh prediction in the wild. In *Computer Vision and Pattern Recognition (CVPR)*, 2021. 3

[67] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4578–4587, 2021. 3

[68] Jingyang Zhang, Yao Yao, and Long Quan. Learning signed distance field for multi-view surface reconstruction. *International Conference on Computer Vision (ICCV)*, 2021. 3

[69] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5745–5753, 2019. 5