OmniShape: Zero-Shot Multi-Hypothesis Shape and Pose Estimation in the Real World

Katherine Liu[1], Sergey Zakharov[1], Dian Chen[1], Takuya Ikeda[2],
Greg Shakhnarovich[3], Adrien Gaidon[1], Rares Ambrus[1]

***Abstract*— We would like to estimate the pose and full shape of an object from a single observation, without assuming known 3D model or category. In this work, we propose OmniShape, the first method of its kind to enable probabilistic pose and shape estimation. OmniShape is based on the key insight that shape completion can be decoupled into two multi-modal distributions: one capturing how measurements project into a normalized object reference frame defined by the dataset and the other modelling a prior over object geometries represented as triplanar neural fields. By training separate conditional diffusion models for these two distributions, we enable sampling multiple hypotheses from the joint pose and shape distribution. OmniShape demonstrates compelling performance on challenging real world datasets. Project website: `https://tri-ml.github.io/omnishape`**

I. Introduction

Detailed understanding of the 3D world is a core challenge in applications ranging from augmented reality to robotics. Despite recent progress in open-world image understanding [1], [2], estimating the complete and accurate 3D geometry of objects in a scene from a single view is an open problem. Consider the case of the cup in Fig. 1 for which a handle has not been observed: in such a case both the shape and pose of the object are uncertain and under-constrained. We are interested in enabling joint multi-hypothesis pose estimation *and* shape completion. Our work is to our knowledge the first to address these two goals jointly, without assuming known geometry or tight constraints on object category.

The vast majority of techniques for **pose estimation** assume object geometry is known *a priori* at an instance or category level. Given an observation of a scene comprised of known object models, the relative poses of the objects can be estimated with approaches such as classical correspondence based methods [3], [4], template matching [5], inverse rendering [6] and learning-based methods [7]. A number of approaches use probabilistic techniques [8]–[10] to deal with pose uncertainty stemming from self-occlusion and symmetry. However, assumptions of known object or limited categories severely limits existing methods' utility in open-world settings. Some recent methods [11], [12] estimate the pose of novel instances, but assume multiple observations.

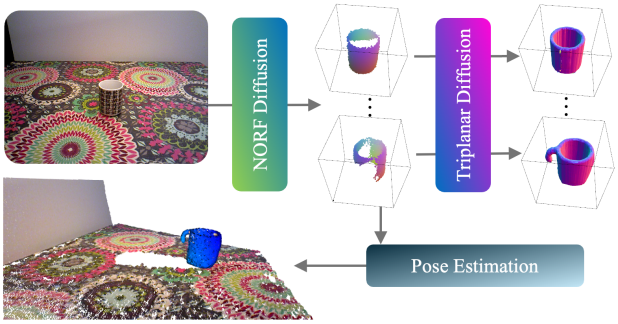

Fig. 1: **OmniShape** decomposes object estimation from an RGB-D image into two stages: estimating partial pointclouds in a normalized object reference frame and conditional shape completion. By using diffusion models for each stage, OmniShape predicts multiple hypotheses of pose and shape, in this case capturing possible object symmetry and/or canonicalization as well as potential geometry under occlusion. The first stage provides per-point association to depth images, enabling registration of the object. Input image from [13].

While single-view **shape completion** methods seek to estimate full 3D object geometry, existing approaches make assumptions that make them difficult or brittle to apply in real-world estimation tasks. For example, many methods use the ShapeNet dataset [14], where instances within a single class are aligned, to learn shape representations [15], [16]. Other methods eschew known canonicalization, learning image-conditioned generative models [17], [18] over large-scale data. However, both such approaches can be challenging to apply in the real world as their internal reference frames are difficult to relate to metric observations. An alternative is to assume objects are observed in identity pose and complete object shapes in the camera coordinate frame, featuring methods including regression [19], [20] and novel-view synthesis [21], [22]. Such approaches define bounds on the extents of the objects for surface extraction, which can be brittle depending on the severity of self-occlusion.

In this work, we propose OmniShape, a novel framework for joint shape completion and pose estimation in the real world. OmniShape is based on the key technical insight that shape completion can be decoupled into two multi-modal distributions. Relaxing the strict dataset canonicalization as-

[1]Toyota Research Institute, Los Altos, CA 94022, USA. {firstname.lastname}@tri.global, †{firstname.lastname}.ctr@tri.global
[2]Woven by Toyota, Chuo City, Tokyo 103-0022, Japan. {firstname.lastname}@woven.toyota
[3]Toyota Technological Institute at Chicago, Chicago, IL 60637, USA. {firstname}@ttic.edu

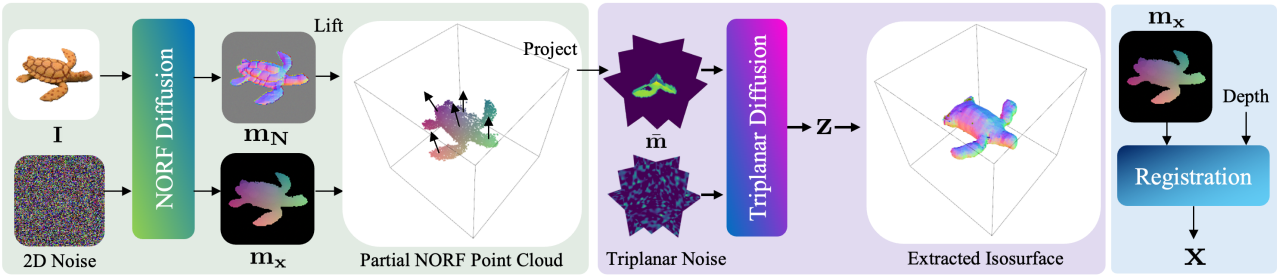

Fig. 2: **System Overview.** OmniShape decouples shape completion into two generative stages. The first (a) maps an RGB (with optional normals from a depth image) to a partial pointcloud with corresponding normals in a NORF. The second (b) conditions on the partial observation in the NORF and predicts the complete object geometry (shown here with CFG), represented as a triplane. Both stages are modeled with diffusion models to produce multiple hypotheses. (c) Given additional information such as a depth image, the object can be registered into the scene using the first stage output. A reduced number of illustrative normals in the NORF pointcloud and channels in the Ortho-NORF conditioning drawn for visual clarity.

sumptions of previous work [8], [23], the first model captures the mapping between images and a Normalized Object Reference Frame (NORF), implicitly capturing pose and partial shape. The second model learns a conditional distribution over complete object shapes, represented as triplanar neural fields. By learning a distribution over NORFs, OmniShape generates partial estimates to condition shape completion in a normalized reference frame, rather than requiring partial measurements to be arbitrarily normalized into a fixed coordinate system. We model the rich multi-modal nature of the distributions via Denoising Diffusion Probabilistic Models [24]. OmniShape outputs pairs of shapes and dense correspondences that enable placing the predicted object into the scene, bridging probabilistic pose estimation and generative shape modeling. We train OmniShape on synthetic data and test on challenging real-world estimation tasks.

In summary, our contributions are as follows: (1) A multi-hypothesis algorithm for jointly estimating the pose and complete shape of objects from a single image, without requiring any prior knowledge about the object (2) A framework that applies the notion of predicting normalized object coordinate spaces [23] from images to "in-the-wild" datasets, relaxing strict canonicalization requirements for use in single-view shape completion (3) An approach to shape completion via diffusion of objects modelled as triplanar grids conditioned on a partial pointcloud observation.

II. Our Approach: OmniShape

We would like to jointly estimate the pose $\mathbf{x} \in SE(3)$ and shape $\mathbf{z}$ (described in Sec. II-C) of an object from a single cropped, segmented RGB-D observation $\mathbf{I} \in \mathbb{R}^{d \times d \times 4}$, where d is the crop resolution. Generally, an object's pose cannot be fully specified without knowledge of the shape, and vice versa. However, the joint conditional probability distribution $p(\mathbf{x}, \mathbf{z}|\mathbf{I})$ can be difficult to sample directly from due to its complex and multi-modal nature.

Inspired by prior work in canonical coordinate estimation [23], we replace $\mathbf{x}$ with an image-like map $\mathbf{m} \in \mathbb{R}^{d \times d \times 3}$ that projects normalized 3D coordinates of visible object points to the camera reference frame – the NORF map (Sec. II-B). The NORF map provides dense pixel to 3D association, enabling the recovery of $\mathbf{x}$ from $\mathbf{m}$ via registration methods given observed depth. We then model the joint probability over object geometry and pose as $p(\mathbf{z}, \mathbf{m}|\mathbf{I})$.

Besides enabling pose estimation, $\mathbf{m}$ *also* provides a partial pointcloud observation of the object surface, from which the object shape can be completed. This is the key insight in OmniShape: we disentangle joint reasoning about pose and shape into a chain of two distributions: (1) the observed surface points in a normalized object reference frame $\mathbf{m}$ given the image $\mathbf{I}$ and (2) the object geometry $\mathbf{z}$ given the partial observation in $\mathbf{m}$:

$$p(\mathbf{z}, \mathbf{m}|\mathbf{I}) = p(\mathbf{z}|\mathbf{m})p(\mathbf{m}|\mathbf{I}), \tag{1}$$

where we assume that $\mathbf{m}$ provides the necessary information to model $\mathbf{z}$. OmniShape approximates both conditional distributions with diffusion models, learning two models ϵ_θ^m and ϵ_θ^z to represent $p(\mathbf{m}|\mathbf{I})$ and $p(\mathbf{z}|\mathbf{m})$ respectively, and enabling sampling from the joint distribution via (1). We highlight that a depth measurement is only needed at test time to estimate a scaled, metric pose.

A. Diffusion Preliminaries

Denoising Diffusion Probabilistic Models (DDPMs) [24] model generative processes by learning to iteratively denoise noisy inputs. Intuitively, diffusion models assume a forward noising process of iteratively adding normally distributed noise to the state $\mathbf{u}$: $q(\mathbf{u}_t|\mathbf{u}_{t-1}) = \mathcal{N}(\sqrt{1-\beta_t}\mathbf{u}_{t-1}, \beta_t\mathbf{I})$, where β_t changes according to a predefined variance schedule. To enable a backwards "denoising" process, a function ϵ_θ can be trained to predict the amount of unscaled noise $\epsilon \sim \mathcal{N}(0, \mathbf{I})$ in a given noisy input $\mathbf{u}_t$, i.e., to minimize a noise matching objective [24]:

$$\mathcal{L}(\theta) = \mathbb{E}_{t, \mathbf{u}_0, \epsilon}\left[\|\epsilon - \epsilon_\theta(\mathbf{u}_t, t)\|^2\right]. \tag{2}$$

Given a trained denoising function, a sample drawn from random noise is then iteratively denoised. Diffusion models

can be further made to model conditional distributions via methods such as classifier guidance [25] or classifier-free guidance (CFG) [26]. In this work, we approximate both $p(\mathbf{z}|\mathbf{m})$ and $p(\mathbf{m}|\mathbf{I})$ with diffusion models, and optionally use classifier free guidance to generate samples from $p(\mathbf{z}|\mathbf{m})$.

B. NORF Diffusion

The Normalized Object Coordinate Space (NOCS) [23] maps pixel coordinates to a normalized reference frame for pose estimation. This requires *canonicalization* – alignment to a shared coordinate system, e.g., defined for a coherent semantic category. However, we are interested in more general scenarios and datasets, where canonicalization cannot be ensured, such as the recently proposed Objaverse dataset [27]. We therefore relax this assumption, and to avoid confusion, name our normalized coordinate framework NORF: Normalized Object Reference Frame. Other methods have also recognized the utility of NOCS-like parameterizations for shape estimation from multi-view images [28] and text [29]; OmniShape utilizes this insight specifically for decoupling multi-hypothesis shape and pose estimation.

OmniShape assumes a dataset of O object models, each contained in a unit cube centered at the origin of the 3D coordinate system. Following [23], we project the visible surface into a posed camera with known intrinsics to obtain a NORF map $\mathbf{m}_x \in \mathbb{R}^{d\times d\times 3}$, an image-like quantity where each pixel value indicates the 3D position in the NORF. We also build a NORF normal map $\mathbf{m}_N \in \mathbb{R}^{d\times d\times 3}$, where each pixel value is the surface normal of the observed point. We construct training tuples of observed images, observed normals, and corresponding output partial NORF maps, i.e., $(\mathbf{I}, \mathbf{N}, \mathbf{m})$, where $\mathbf{m}_x$ and $\mathbf{m}_N$ comprise the NORF measurement $\mathbf{m}$. As in previous work [8] we use normal $\mathbf{N}$, which can be calculated from depth, to avoid brittleness in normalizing arbitrary depth measurements. We train the DDPM as in (2), with the NORF map $\mathbf{m}$ as state, and the observed RGB image $\mathbf{I}$ and normals $\mathbf{N}$ as the conditoning:

$$\mathcal{L}(\theta) = \mathbb{E}_{t,(\mathbf{m}_0,\mathbf{I},\mathbf{N}),\epsilon}\left[\|\epsilon - \epsilon^m_\theta(\mathbf{m}_t, t, \mathbf{I}, \mathbf{N})\|^2\right]. \quad (3)$$

We use ϵ^m_θ to approximate conditional sampling of partial pointcloud observations $p(\mathbf{m}|\mathbf{I})$, illustrated in Fig. 2a.

Although OmniShape does not enforce strict intra-class canonicalization, our experiments show the sampled NORF maps trained on human generated or processed datasets exhibit evidence of structure. Fig. 3 shows cup openings pointing upwards and the axis aligned airplanes, suggesting some inherent alignment rules. The canonicalization and therefore pose distribution modeled is a function of the dataset. OmniShape uses these patterns to avoid optimizing triplanes online for objects in arbitrary poses as shapes can be completed in common reference frames learned from data.

C. Triplanar Field Diffusion

We model objects as signed distance fields, represented by triplanar neural fields [30]. Each object is represented by a triplanar latent $\mathbf{z} \in \mathbb{R}^{3\times 2^p\times 2^p\times n}$, where n is the dimension of the latent and p is the level of detail. The triplanar representation allows for continuous neural fields to be represented via three orthogonal $2^p \times 2^p$ feature planes. To query for the signed distance of an arbitrary point $\mathbf{p} \in \mathbb{R}^3$, we project the coordinate onto three orthogonal planes. We then perform interpolation per plane and concatenate the resulting features to obtain the latent for the coordinate, i.e., $\bar{z}_\mathbf{p} = \omega(\mathbf{p}, \mathbf{z})$, where $\bar{z} \in \mathbb{R}^{3n}$. To obtain the final signed distance value $\mathbf{f_p}$, we learn a decoder ξ such that $\mathbf{f_p} = \xi(\bar{z}_\mathbf{p})$.

For each object, we assume a SDF pointcloud tuple $\{(\mathbf{s}^i_0, d^i_0)...(\mathbf{s}^i_{M_i}, d^i_{M_i})\}$, which pairs M_i sampled 3D points $\mathbf{s} \in \mathbb{R}^3$ coupled with their distance $d \in \mathbb{R}$ from the surface of the i-th object. We formulate an optimization over the set of triplanar latents $\mathcal{Z} = \{\mathbf{z}_0, ..., \mathbf{z}_O\}$ as well as the set of parameters of the decoder ξ to minimize the L1 reconstruction loss, combined with a total variation (TV) term summed over each of the three feature planes as in [31]:

$$\mathcal{L}(\mathcal{Z}, \xi) = \sum_{i=0}^{i=O}\sum_{j=0}^{j=M_i} \left|\xi(\omega(\mathbf{s}^i_j, \mathbf{z}_i)) - d^i_j\right| + \alpha_{TV}\sum_{i=0}^{i=O}\mathrm{TV}(\mathbf{z}_i). \quad (4)$$

After a set of triplanes has been optimized, we use pairs of optimized triplanes and partial pointclouds with normals in the NORF to train the shape completion diffusion model. The triplanar representation can be rearranged into image-like tensors of dimension $\mathbf{z}' \in \mathbb{R}^{2^p\times 2^p\times 3n}$, allowing the use of 2D diffusion methods for the shape completion model ϵ^z_θ. Importantly, the shape completion process can assume that NORF predictions are *already* in a normalized reference frame, avoiding brittle pre-prediction normalization.

We find it is important at the shape completion stage to align the NORF predictions described in Sec. II-B to the trained triplanes. To this end, we voxelize $\mathbf{m}$ into an occupancy grid with side dimension 2^{p+1} (keeping the average normal value for occupied cells), then orthogonally project the values onto the three planes, generating measurements aligned with the triplanar representation. We then perform Pixel Unshuffling [32] to reduce the spatial dimension of the conditioning by half. This process builds the Ortho-NORF $\overline{\mathbf{m}} \in \mathbb{R}^{2^p\times 2^p\times 48}$: the partial pointcloud with normals, orthogonally projected into the triplanar space. At inference time, we filter noisy points from predicted $\mathbf{m}$ before projection.

We then adapt Eq. 2 for shape completion diffusion:

$$\mathcal{L}(\theta) = \mathbb{E}_{t,(\mathbf{z}',\overline{\mathbf{m}}),\epsilon_t}\left[\|\epsilon_t - \epsilon^z_\theta(\mathbf{z}'_t, t, \overline{\mathbf{m}})\|^2\right], \quad (5)$$

where the state is a triplane $\mathbf{z}'$ from $\mathcal{Z}$ and the conditioning is $\overline{\mathbf{m}}$. At inference time, we sample from $p(\mathbf{z}|\mathbf{m})$ using the trained ϵ^z_θ. Fig. 2b illustrates this process.

D. Shape Completion and Registration

Given an RGB-D image with an object segmentation mask at inference time, OmniShape uses the diffusion model ϵ^m_θ to sample $\mathbf{m}$, which is used as conditioning to sample object completions from the trained model ϵ^z_θ. From the sampled $\mathbf{m}$ and $\mathbf{z}$, an explicit pose $\mathbf{x}$ and surface can be extracted, enabling the object to be placed into the scene. Multiple hypotheses from the joint distribution of pose and

shape can be generated by applying the iterative sampling process multiple times. Registration is enabled by the per-pixel association between a point in the NORF and the observed image given by $\mathbf{m}$. We show in Sec. III that a pose registration metric can also be a practical hypothesis selection method.

III. Experiments

To evaluate OmniShape, we test performance on several challenging object-centric estimation datasets.

A. Implementation Details

Data. We train OmniShape on the dataset proposed in ZeroShape [19], featuring 84789 objects with corresponding SDF supervision and over 1 million corresponding renderings sourced from ShapeNet [14] and the Objaverse-LVIS split [27]. While ϵ_θ^m is trained with optional input normals, for all evaluation results we use only RGB images.

Triplane Model. For our experiments, we use triplanes with $p = 5$ and $n = 12$, use a MLP with layers of width $(36, 512, 512, 1)$ and Leaky ReLU activations to decode interpolated latent values into signed distances. We set $\alpha_{TV} = 0.01$ and randomly sample 5k supervision points per object each epoch. The network is trained on eight NVIDIA A100s with a batch size of 128 per GPU and learning rate of 0.01 for 9299 epochs. We use octree subdivision (similar to [33]) to a level of detail 6 to extract points on the isosurface. Before input to ϵ_θ^z we normalize the triplanes per channel to have a standard deviation of 0.2, and clip values to $[-1, 1]$.

TABLE I: **Single-object reconstruction results.** *Ours, FH* indicates results when taking the first hypothesis generated. *Ours, Best-of-N* indicates taking the best result compared to the ground-truth according to the Chamfer distance over N hypotheses, which we report to quantify the quality of the best hypothesis generated. We report our metrics as (without CFG/with CFG=5). OmniShape's first hypothesis is competitive with the baseline methods, while taking additional hypotheses results in better solutions existing in the hypothesis set indicating the benefits of our probabilistic formulation. All baseline metrics taken from ZeroShape [19].

	Ocrtoc3D		Pix3D	
Method	FS@1↑	CD↓	FS@1↑	CD↓
SS3D [34]	0.1271	0.543	0.1326	0.485
MCC [20]	0.1994	0.411	0.1754	0.514
One-2-3-45 [35]	0.1323	0.492	0.1364	0.443
OpenLRM [36]	0.1552	0.432	0.1458	0.492
Shap-E [37]	0.1725	0.395	**0.2016**	**0.340**
ZeroShape [19]	**0.2410**	**0.286**	0.1928	0.345
Ours, FH	0.1952/0.1975	0.376/0.367	0.1675/0.1704	0.426/0.426
Ours, best-of-5	0.2477/0.2491	0.272/0.268	0.2214/0.2200	0.318/0.326
Ours, best-of-25	0.2856/**0.2867**	0.233/**0.230**	**0.2622**/0.2571	**0.263**/0.272

Diffusion Models. For ϵ_θ^m, we use square crops of dimension 128 and a 2D UNet [38] implementation [39] with channel dimensions of $(128, 128, 256, 256, 512, 512)$, with the fifth layer of the encoder and second layer of the decoder as attention blocks. For ϵ_θ^z, we use a modified version of different UNet variant [25], with channel dimensions $(540, 1080, 2160)$ and attention at resolutions 2,4. Both networks assume a linear noise schedule with 1000 steps and beta ranging from 0.0001 to 0.02, and train using cosine learning rate decay with 500 warm up steps and peak learning rate of 0.0001 on eight NVIDIA A100s. ϵ_θ^m is trained with a batch size of 128 per GPU for 1000 epochs, while ϵ_θ^z is trained with a batch size of 32 per GPU for 100 epochs.

Conditioning is implemented via channel concatenation, except for timestep conditioning which depends on the implementations noted above. When training ϵ_θ^m, we downscale-upscale with 25% probability, rotate input and output with probability 50%, and drop normal conditioning with probability 50%; for ϵ_θ^z we drop all conditioning with probability 20%. We do not use CFG except where noted in Tab. I paired only with ϵ_θ^z, experimentally finding that CFG benefits performance on only some test sets. For inference, we use the DPM-Solver++ [40] with 50 steps for $\mathbf{m}$ and 25 steps for $\mathbf{z}$. Without batching or CFG, generating a single Ortho-NORF estimate takes approximately 2.1s seconds and a single shape completion (including surface extraction) takes approximately 0.9s on a NVIDIA RTX A6000.

Baselines. We compare to several pre-trained zeroshot image-to-3D methods, both deterministic: SS3D [34], MCC [20], One-2-3-45 [35], LRM [41]/OpenLRM [36], ZeroShape [19] and stochastic: Shap-E [37]. Object representations include NERFs [34], [35], [41], pointclouds [20], and occupancy fields [19]. SS3D learns a shape space using ShapeNet [14], then utilizes image datasets via a multi-hypothesis camera approach. One-2-3-45 leverages a novel view synthesis diffusion model [42] to inform shape completion. OpenLRM is an implementation of LRM [41], a transformer based method which predicts triplanar NeRFs trained on Objaverse. ZeroShape predicts depth and intrinsics, then normalizes the observed surface from which the object is completed. MCC is an encoder-decoder method trained on CO3D [43]. Shap-E is a diffusion-based method trained on millions of 3D objects [44] that predicts objects in an internal (unknown) reference frame. ZeroShape and OmniShape predict only geometry and are the only methods to share the same training split.

B. Single-Object Reconstruction

We evaluate shape completion performance on two real world datasets of single objects, using the processed versions provided by ZeroShape [19]: Ocrtoc3D [49], consisting of 749 images of diverse objects captured in the wild, and Pix3D [50], featuring 1181 images of primarily furniture. We report Chamfer distance and F1 score following the protocol described in [19], finding the lowest Chamfer distance over a discrete set of rotations, using 10k points from the estimate.

Quantitative results are given in Tab. I and selected qualitative results in Fig. 3. Baseline values (all except ours) are taken from [19]. Given the first hypothesis, OmniShape exhibits performance similar to several state of the art methods. Taking the best estimate of 5 OmniShape hypotheses (as determined by comparing to the ground-truth) our method outperforms the other methods and further improves with 25

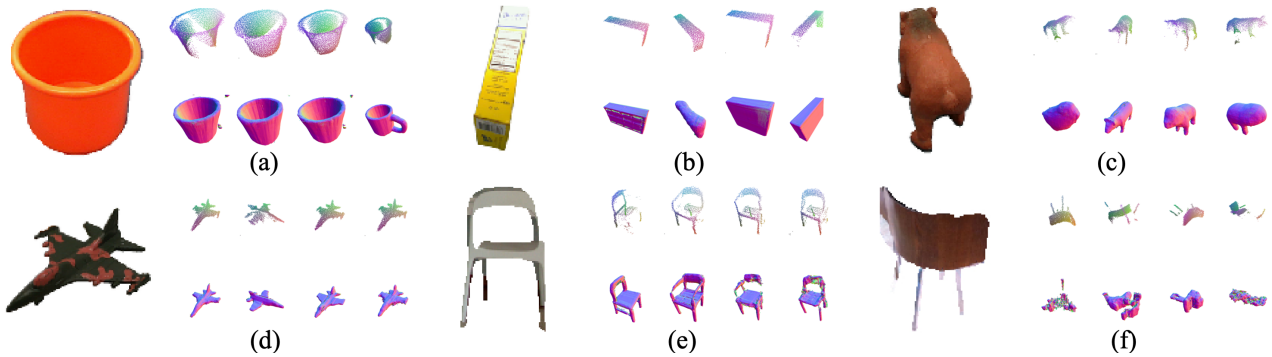

Fig. 3: **Qualitative shape completion results.** Given an RGB image, OmniShape estimates multiple hypotheses for both the partial pointcloud in the NORF and the corresponding shape completion. For each input image, we show four partially filtered partial pointcloud predictions (top row) and one conditional shape completion each (bottom row). The probabilistic nature of OmniShape enables the prediction of different potential shape completions, useful under ambiguity. For example, OmniShape predicts cups of different geometries in (a), boxes of different dimensions in (b), and different animals in (c). In (f) we show an example failure case exhibiting incoherent geometry. (a)-(d) from Ocrtoc3D and (e)-(f) from Pix3D.

TABLE II: **Real-world results**. On the challenging task of estimating the shape of objects and registering the objects in real-world coordinates, OmniShape predictions paired with inlier selection over 10 hypotheses outperforms Zeroshape on TYO-L and NOCS as measured by L1 Chamfer Distance and F1 score (threshold of 5). Taking multiple OmniShape hypotheses generates estimates with higher accuracy across all three datasets. Due to high variance when taking the average over all objects, we instead report the mean and standard deviation over the per-scene means, bolding results based on mean value.

	TYO-L (21 scenes, 1670 objects in total)		**NOCS** (6 scenes, 875 objects in total)		**HOPE** (10 scenes, 920 objects in total)	
Method	CD↓	F1↑	CD↓	F1↑	CD↓	F1↑
Zeroshape	13.35±5.36	0.31±0.10	14.57±2.82	0.30±0.02	**9.94**±4.48	0.49±0.06
Ours-RGB, first hypothesis	10.98±4.89	0.47±0.11	15.45±2.26	0.35±0.05	13.35±10.76	0.47±0.06
Ours-RGB, inlier selection	**8.81**±4.04	**0.54**±0.11	**10.67**±1.22	**0.42**±0.04	10.36±7.64	**0.54**±0.07
Ours-RGB, best-of-10	5.76±2.56	0.65±0.13	7.73±0.87	0.51±0.05	7.04±3.90	0.64±0.08

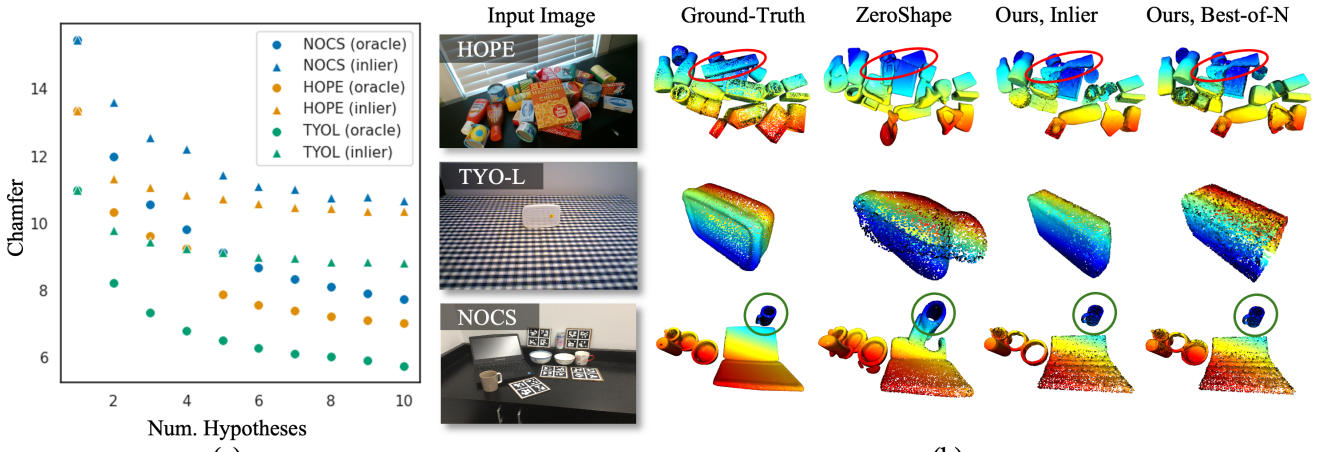

Fig. 4: **Results on shape and pose estimation.** (a) We show that sampling additional hypotheses reduces the Chamfer distance on all three datasets, with the inlier-based method plateauing earlier than the oracle-based method. (b) Qualitative visualizations of performance on the datasets. We show the input image as well as the groundtruth and reconstruction by various methods. The middle example (from TYO-L) highlights a key benefit of the multi-hypothesis nature of our method; OmniShape can predict shapes of varying sizes given an ambiguous view of only the front of the container. We also observe that although neither ZeroShape nor OmniShape are trained to handle occlusions, OmniShape can sometimes fill in missing geometry as in the cup on the bottom row (green circle) and the thin spaghetti box on the top row (red ellipse). Training time image augmentations may also help OmniShape handle images encountered in the wild.

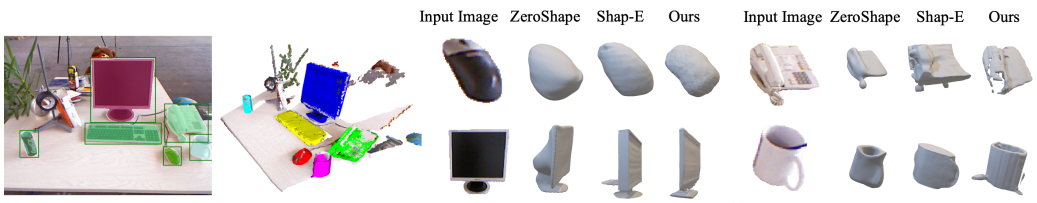

Fig. 5: **Qualitative example of shape and pose estimation.** We use the inlier hypothesis selection method over 25 OmniShape hypotheses to complete objects in an image from the TUM dataset [45]. (a) We detect [46] and segment [47] several objects from the scene and show (b) OmniShape meshes (extracted via a differentiable watertight mesh extractor [48] and decimated) and estimated NORF points after being registered into the scene, overlaid on the input pointcloud. In (c), we show qualitative performance of OmniShape, ZeroShape and Shap-E. Each method estimates objects in a different reference frame; we manually rotate to provide approximate representative views. Videos of meshes are provided in the attached multimedia.

hypotheses. Although the best-of-N metric cannot be used in online inference, these results demonstrate the usefulness of a multi-hypothesis method. Furthermore, unlike the other generative method considered [37] that predicts objects in an unknown coordinate system, OmniShape enables the predicted shape to be estimated in world coordinates.

C. Real-World Reconstruction and Pose Estimation

We evaluate OmniShape for object shape and pose estimation on three real-world datasets, using the provided images, depths, segmentations, and meshes. Image inputs are interpolated via a nearest strategy to the size required by each method. TYO-L [13] features one object per scene with varied poses/lighting; we keep a max of the first 80 images from each of the 21 scenes. NOCS [23] includes multiple objects such as mugs and laptops; we keep the first 25 images from each of the 6 scenes in the REAL275 test set. HOPE [51] includes crowded toy food scenes; we keep the first 5 images from each of the 10 scenes. These benchmarks are typically used to evaluate pose estimation with strictly consistent object canoncalizations between train and test. OmniShape relaxes strict canonicalization requirements but therefore cannot directly measure object pose error compared to an arbitrary canonical frame defined by any one benchmark. Instead, we measure performance via Chamfer-L1 and F1 score after object registration into the scene, sampling 10k points from estimated objects.

We compare to ZeroShape [19], one of the best performing baselines from Tab. I that also provides a straight-forward method of estimating the object pose in metric coordinates. Procrustes and RANSAC are used to align the observed depth and prediction. For ZeroShape we solve for the alignment between the predicted pointcloud from its first stage to the points observed from depth, while for OmniShape we use the estimated $\mathbf{m}_x$, which provides the 3D NORF coordinate of a given pixel. We use the same registration settings for both methods, including resizing output predictions to a consistent dimension. We generate 10 OmniShape hypotheses, selecting the hypothesis with the most inliers after registration. We also report best-of-N results determined by the groundtruth to quantify the quality of the best hypothesis generated. Metrics are calculated after placing the estimates into the scene, first taking the mean over all objects in all images from a single scene, then the mean and standard deviation over the per-scene results.

Results are shown in Tab. II and Fig. 4. Although ZeroShape exhibits good generalization performance, it can struggle when the geometry of the object is occluded, likely due to its determinism. Using the number of inliers to select the best OmniShape hypothesis yields better performance on average on TYO-L and NOCS. Fig. 4a shows the relationship of taking more OmniShape hypotheses, illustrating the utility of our multi-hypothesis method. Using the number of inliers for hypothesis selection does not match the performance of an oracle. This is unsurprising as the registration-based metric depends only on the visible portion of the object and not the quality of the shape completion, which can result in low quality hypothesis selection. Nevertheless, the best-of-N results show our method produces more accurate estimates on average among multiple hypotheses over all three datasets.

Finally, Fig. 5 provides a qualitative example of shape and pose estimation on a real RGB-D image without groundtruth. While our results are inherently stochastic and can vary, the qualitative results show that compared to the deterministic ZeroShape which predicts overly smooth geometries and the generative Shap-E which can predict different geometries but does not have a straight-forward method of selecting the best hypothesis or pose estimation, OmniShape is a generative method that enables pose estimation as seen in Fig. 5(b).

IV. Conclusion and Limitations

We have presented OmniShape, a method for shape and pose estimation in the real world. OmniShape uses diffusion models to first predict a partial pointcloud in a Normalized Object Reference Frame, and then to predict the shape completion. We demonstrate that OmniShape's multi-hypothesis nature can lead to higher geometric accuracy considering best-of-N metrics, and the number of inliers from a registration process can be a useful signal for hypothesis selection.

Despite OmniShape's promising results, limitations still exist. OmniShape can struggle on objects with very detailed

geometry and also predict incoherent surfaces (Fig. 3f). In practice, noisy real-world normals also appear out of distribution from synthetic training normals and can harm predictions. Future directions include using semantic feature [17], [52] conditioning, finetuning on zero-shot normal estimates [53], and learning a hypothesis selection metric.

ACKNOWLEDGMENT

Code generation tools [54] were used in the course of developing the code for this work.

REFERENCES

[1] X. Gu, T.-Y. Lin, W. Kuo, and Y. Cui, "Open-vocabulary object detection via vision and language knowledge distillation," *arXiv preprint arXiv:2104.13921*, 2021.

[2] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W. Lo, P. Dollár, and R. B. Girshick, "Segment anything," in *Proc. IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE, 2023.

[3] S. Umeyama, "Least-squares estimation of transformation parameters between two point patterns," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 13, no. 04, pp. 376–380, 1991.

[4] X.-S. Gao, X.-R. Hou, J. Tang, and H.-F. Cheng, "Complete solution classification for the perspective-three-point problem," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 8, pp. 930–943, 2003.

[5] V. N. Nguyen, T. Groueix, M. Salzmann, and V. Lepetit, "GigaPose: Fast and robust novel object pose estimation via one correspondence," *arXiv preprint arXiv:2311.14155*, 2023.

[6] Y. Lin, P. Florence, J. T. Barron, A. Rodriguez, P. Isola, and T. Lin, "iNeRF: Inverting neural radiance fields for pose estimation," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems, (IROS)*. IEEE, 2021.

[7] S. Zakharov, I. Shugurov, and S. Ilic, "DPOD: 6D pose object detector and refiner," in *Proc. IEEE/CVF International Conference on Computer Vision (CVPR)*, 2019, pp. 1941–1950.

[8] T. Ikeda, S. Zakharov, T. Ko, M. Z. Irshad, R. Lee, K. Liu, R. Ambrus, and K. Nishiwaki, "DiffusionNOCS: Managing symmetry and uncertainty in sim2real multi-modal category-level pose estimation," *arXiv preprint arXiv:2402.12647*, 2024.

[9] X. Deng, A. Mousavian, Y. Xiang, F. Xia, T. Bretl, and D. Fox, "PoseRBPF: A Rao–Blackwellized particle filter for 6D object pose tracking," *IEEE Transactions on Robotics*, vol. 37, no. 5, pp. 1328–1342, 2021.

[10] J. Zhang, M. Wu, and H. Dong, "Generative category-level object pose estimation via diffusion models," *Advances in Neural Information Processing Systems (NeuRIPS)*, vol. 36, 2024.

[11] K. Park, A. Mousavian, Y. Xiang, and D. Fox, "LatentFusion: End-to-end differentiable reconstruction and rendering for unseen object pose estimation," in *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 10 710–10 719.

[12] B. Wen, J. Tremblay, V. Blukis, S. Tyree, T. Müller, A. Evans, D. Fox, J. Kautz, and S. Birchfield, "BundleSDF: Neural 6-DoF tracking and 3D reconstruction of unknown objects," in *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 606–617.

[13] T. Hodan, F. Michel, E. Brachmann, W. Kehl, A. GlentBuch, D. Kraft, B. Drost, J. Vidal, S. Ihrke, X. Zabulis *et al.*, "BOP: Benchmark for 6D object pose estimation," in *Proc. European Conference on Computer Vision (ECCV)*, 2018, pp. 19–34.

[14] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, J. Xiao, L. Yi, and F. Yu, "ShapeNet: An Information-Rich 3D Model Repository," Stanford University — Princeton University — Toyota Technological Institute at Chicago, Tech. Rep. arXiv:1512.03012 [cs.GR], 2015.

[15] J. J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove, "DeepSDF: Learning continuous signed distance functions for shape representation," in *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 165–174.

[16] Y.-C. Cheng, H.-Y. Lee, S. Tulyakov, A. Schwing, and L. Gui, "SDFusion: Multimodal 3D Shape Completion, Reconstruction, and Generation," *arXiv preprint arXiv:2212.04493*, 2022.

[17] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *Proc. International Conference on Machine Learning (ICML)*. PMLR, 2021, pp. 8748–8763.

[18] A. Nichol, H. Jun, P. Dhariwal, P. Mishkin, and M. Chen, "Point-E: A system for generating 3D point clouds from complex prompts," *arXiv preprint arXiv:2212.08751*, 2022.

[19] Z. Huang, S. Stojanov, A. Thai, V. Jampani, and J. M. Rehg, "ZeroShape: Regression-based zero-shot shape reconstruction," in *In Proc. 2024 Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 10 061–10 071.

[20] C.-Y. Wu, J. Johnson, J. Malik, C. Feichtenhofer, and G. Gkioxari, "Multiview Compressive Coding for 3D Reconstruction," in *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 9065–9075.

[21] M. Liu, R. Shi, L. Chen, Z. Zhang, C. Xu, X. Wei, H. Chen, C. Zeng, J. Gu, and H. Su, "One-2-3-45++: Fast single image to 3d objects with consistent multi-view generation and 3d diffusion," *2024 Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10 072–10 083, 2023.

[22] J. Tang, J. Ren, H. Zhou, Z. Liu, and G. Zeng, "DreamGaussian: Generative gaussian splatting for efficient 3D content creation," *arXiv preprint arXiv:2309.16653*, 2023.

[23] H. Wang, S. Sridhar, J. Huang, J. Valentin, S. Song, and L. J. Guibas, "Normalized object coordinate space for category-level 6D object pose and size estimation," in *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2642–2651.

[24] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, pp. 6840–6851, 2020.

[25] P. Dhariwal and A. Nichol, "Diffusion models beat GANS on image synthesis," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 34, pp. 8780–8794, 2021.

[26] J. Ho and T. Salimans, "Classifier-free diffusion guidance," *arXiv preprint arXiv:2207.12598*, 2022.

[27] M. Deitke, D. Schwenk, J. Salvador, L. Weihs, O. Michel, E. VanderBilt, L. Schmidt, K. Ehsani, A. Kembhavi, and A. Farhadi, "Objaverse: A universe of annotated 3D objects," in *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 13 142–13 153.

[28] C. Xu, A. Li, L. Chen, Y. Liu, R. Shi, H. Su, and M. Liu, "SpaRP: Fast 3d object reconstruction and pose estimation from sparse views," *18th European Conference on Computer Vision (ECCV), Milano, Italy.*, 2024.

[29] W. Li, R. Chen, X. Chen, and P. Tan, "SweetDreamer: Aligning geometric priors in 2D diffusion for consistent text-to-3D," *arXiv preprint arXiv:2310.02596*, 2023.

[30] E. R. Chan, C. Z. Lin, M. A. Chan, K. Nagano, B. Pan, S. De Mello, O. Gallo, L. J. Guibas, J. Tremblay, S. Khamis *et al.*, "Efficient geometry-aware 3D generative adversarial networks," in *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 16 123–16 133.

[31] J. R. Shue, E. Chan, R. Po, Z. Ankner, J. Wu, and G. Wetzstein, "3D neural field generation using triplane diffusion," *Proc. IEEE/CVF Conference on Computer Vision (CVPR)*, pp. 20 875–20 886, 2023.

[32] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 1874–1883.

[33] M. Z. Irshad, S. Zakharov, R. Ambrus, T. Kollar, Z. Kira, and A. Gaidon, "SHAPO: Implicit representations for multi-object shape, appearance, and pose optimization," in *Proc. European Conference on Computer Vision (ECCV)*. Springer, 2022, pp. 275–292.

[34] K. V. Alwala, A. Gupta, and S. Tulsiani, "Pre-train, self-train, distill: A simple recipe for supersizing 3D reconstruction," in *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 3773–3782.

[35] M. Liu, C. Xu, H. Jin, L. Chen, M. Varma T, Z. Xu, and H. Su, "One-2-3-45: Any single image to 3D mesh in 45 seconds without per-shape optimization," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[36] Z. He and T. Wang, "OpenLRM: Open-source large reconstruction models," https://github.com/3DTopia/OpenLRM, 2023.

[37] H. Jun and A. Nichol, "Shap-E: Generating conditional 3D implicit functions," *arXiv preprint arXiv:2305.02463*, 2023.

[38] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," *ArXiv*, vol. abs/1505.04597, 2015.

[39] P. von Platen, S. Patil, A. Lozhkov, P. Cuenca, N. Lambert, K. Rasul, M. Davaadorj, D. Nair, S. Paul, W. Berman, Y. Xu, S. Liu, and T. Wolf, "Diffusers: State-of-the-art diffusion models," https://github.com/huggingface/diffusers, 2022.

[40] C. Lu, Y. Zhou, F. Bao, J. Chen, C. Li, and J. Zhu, "DPM-solver++: Fast solver for guided sampling of diffusion probabilistic models," *arXiv preprint arXiv:2211.01095*, 2022.

[41] Y. Hong, K. Zhang, J. Gu, S. Bi, Y. Zhou, D. Liu, F. Liu, K. Sunkavalli, T. Bui, and H. Tan, "LRM: Large reconstruction model for single image to 3D," *arXiv preprint arXiv:2311.04400*, 2023.

[42] R. Liu, R. Wu, B. Van Hoorick, P. Tokmakov, S. Zakharov, and C. Vondrick, "Zero-1-to-3: Zero-shot one image to 3D object," in *Proc. IEEE/CVF International Conference on Computer Vision (CVPR)*, 2023, pp. 9298–9309.

[43] J. Reizenstein, R. Shapovalov, P. Henzler, L. Sbordone, P. Labatut, and D. Novotny, "Common objects in 3D: Large-scale learning and evaluation of real-life 3D category reconstruction," in *International Conference on Computer Vision*, 2021.

[44] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "NeRF: Representing scenes as neural radiance fields for view synthesis," *Communications of the ACM*, vol. 65, no. 1, pp. 99–106, 2021.

[45] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A Benchmark for the Evaluation of RGB-D SLAM Systems," in *Proc. International Conference on Intelligent Robot Systems (IROS)*, Oct. 2012.

[46] S. Liu, Z. Zeng, T. Ren, F. Li, H. Zhang, J. Yang, C. Li, J. Yang, H. Su, J. Zhu *et al.*, "Grounding DINO: Marrying DINO with grounded pre-training for open-set object detection," *arXiv preprint arXiv:2303.05499*, 2023.

[47] N. Ravi, V. Gabeur, Y.-T. Hu, R. Hu, C. Ryali, T. Ma, H. Khedr, R. Rädle, C. Rolland, L. Gustafson, E. Mintun, J. Pan, K. V. Alwala, N. Carion, C.-Y. Wu, R. Girshick, P. Dollár, and C. Feichtenhofer, "SAM 2: Segment anything in images and videos," *arXiv preprint arXiv:2408.00714*, 2024.

[48] X. Wei, F. Xiang, S. Bi, A. Chen, K. Sunkavalli, Z. Xu, and H. Su, "NeuManifold: Neural watertight manifold reconstruction with efficient and high-quality rendering support," *arXiv preprint arXiv:2305.17134*, 2023.

[49] R. Shrestha, S. Hu, M. Gou, Z. Liu, and P. Tan, "A real world dataset for multi-view 3D reconstruction," in *Proc. European Conference on Computer Vision (ECCV)*. Springer, 2022, pp. 56–73.

[50] X. Sun, J. Wu, X. Zhang, Z. Zhang, C. Zhang, T. Xue, J. B. Tenenbaum, and W. T. Freeman, "Pix3D: Dataset and methods for single-image 3d shape modeling," in *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 2974–2983.

[51] S. Tyree, J. Tremblay, T. To, J. Cheng, T. Mosier, J. Smith, and S. Birchfield, "6-DoF pose estimation of household objects for robotic manipulation: An accessible dataset and benchmark," in *Proc. International Conference on Intelligent Robots and Systems (IROS)*, 2022.

[52] M. Oquab, T. Darcet, T. Moutakanni, H. V. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, R. Howes, P.-Y. Huang, H. Xu, V. Sharma, S.-W. Li, W. Galuba, M. Rabbat, M. Assran, N. Ballas, G. Synnaeve, I. Misra, H. Jegou, J. Mairal, P. Labatut, A. Joulin, and P. Bojanowski, "DINOv2: Learning robust visual features without supervision," *arXiv:2304.07193*, 2023.

[53] G. Bae and A. J. Davison, "Rethinking inductive biases for surface normal estimation," in *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[54] OpenAI, "ChatGPT: AI Language Model," https://chat.openai.com/.