ShAPO: Implicit Representations for Multi-Object Shape, Appearance, and Pose Optimization

Muhammad Zubair Irshad*[1] , Sergey Zakharov*[2] , Rares Ambrus[2] ,
Thomas Kollar[2] , Zsolt Kira[1] , and Adrien Gaidon[2]

* denotes equal contribution

[1] Georgia Institute of Technology
[2] Toyota Research Institute

[1]{mirshad7, zkira}@gatech.edu, [2]{first.last}@tri.global

Input Image   Detection Heatmaps   3D Bounding Boxes

Implicit Shapes   6D Pose and 3D Detections   Textured Reconstructions
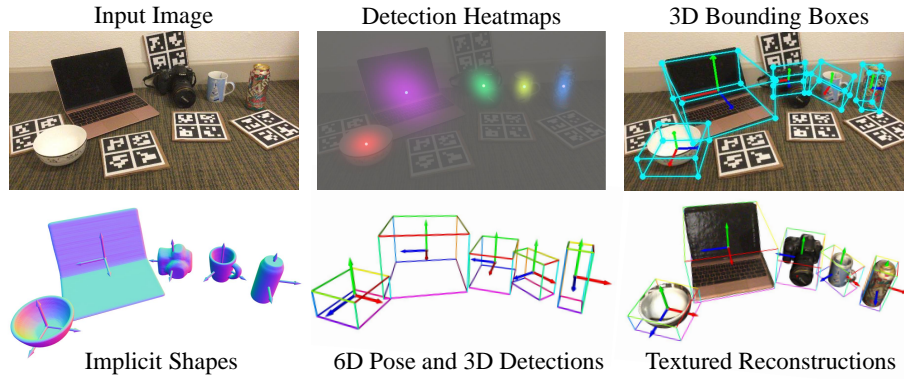

Fig. 1: Given a single RGB-D observation, our method, **ShAPO**, infers 6D pose and size, 3D shapes and appearance of all objects in the scene. Results shown on novel real-world scene in NOCS [54].

Abstract. Our method studies the complex task of object-centric 3D understanding from a single RGB-D observation. As it is an ill-posed problem, existing methods suffer from low performance for both 3D shape and 6D pose and size estimation in complex multi-object scenarios with occlusions. We present ShAPO, a method for joint multi-object detection, 3D textured reconstruction, 6D object pose and size estimation. Key to ShAPO is a single-shot pipeline to regress shape, appearance and pose latent codes along with the masks of each object instance, which is then further refined in a sparse-to-dense fashion. A novel disentangled shape and appearance database of priors is first learned to embed objects in their respective shape and appearance space. We also propose a novel, octree-based differentiable optimization step, allowing us to further improve object shape, pose and appearance simultaneously under the learned latent space, in an analysis-by-synthesis fashion. Our novel joint implicit textured object representation allows us to accurately identify and reconstruct novel unseen objects without having access to their 3D meshes. Through extensive experiments, we show that

our method, trained on simulated indoor scenes, accurately regresses the shape, appearance and pose of novel objects in the real-world with minimal fine-tuning. Our method significantly out-performs all baselines on the NOCS dataset with an 8% absolute improvement in mAP for 6D pose estimation.

Keywords: Implicit Representations; 3D Shape and Texture Reconstruction; 6D Pose Estimation; Octree-based Differentiable Optimization

1 Introduction

Holistic 3D object understanding [31,60] from a single RGB-D observation has remained a popular yet challenging problem in computer vision [11,14,61] and robotics [9,16,19,25]. The capability to infer complete object-centric 3D scene information has benefits in robotic manipulation [6,16,19,25], navigation [17,18] and augmented reality. This task demands that the autonomous robot reasons about the 3D geometry of objects from partially observed single-view visual data, infer a 3D shape and 6D pose and size (i.e. 3D orientation and position) and estimate the appearance of novel object instances. Despite recent progress, this problem remains challenging since inferring 3D shape from images is an ill-posed problem and predicting the 6D pose and 3D scale can be extremely ambiguous without having any prior information about the objects of interest.

Prior works on object-centric scene understanding have attempted to address this challenge in various ways: *object pose understanding* methods obtain 3D bounding box information without shape details. Most prior works in object-pose estimation cast it as an instance-level [21,41,57] 3D object understanding task as opposed to category-level. Such methods [39,50,53], while achieving impressive results, rely on provided 3D reconstructions or prior CAD models for successful detection and pose estimation. Category-level approaches [47,51,54], on the other hand, rely on learned shape and scale priors during training, making them much more challenging. Despite great progress in category-level pose estimation, the performance of these approaches is limited due to their incapacity to express shape variations explicitly. *Object-centric scene-level reconstruction* methods [20,33,55] recover object shapes using 2D or partial 3D information for scene reconstruction. However, most methods are limited in their ability to reconstruct high quality shapes in a fast manner (i.e. the studied representation is either voxel-based which is computationally inefficient [13] or point-based which results in poor reconstruction quality [5,8]).

For holistic scene-level reconstruction, only predicting shapes in an isolated manner [13,24,58] may not yield good results due to the challenges of aligning objects in the 3D space, reasoning about occlusions and diverse backgrounds. To the best our knowledge, fewer works have tackled the problem of joint shape reconstruction with appearance and object-centric scene context (i.e. 3D bounding boxes and object pose and sizes) for a holistic object-centric scene understanding.

Motivated by the above, we present ShAPO: Implicit Representations for Multi-Object Shape, Appearance, and Pose Optimization, an end-to-end method

unifying accurate shape prediction and alignment with object-centric scene context. As shown in Figure 1, we infer the complete 3D information of novel object instances (i.e. 3D shape along with appearance and 6D pose and sizes) from a single-view RGB-D observation; the results shown in Figure 1 are on a novel scene from the NOCS [54] dataset. In essence, our method represents object instances as center key-points [7, 16, 25] in a spatial 2D grid. We regress the complete 3D information i.e. object shape and appearance codes along with the object masks and 6D pose and sizes at each objects' spatial center point. A novel joint implicit shape and appearance database of signed distance and texture field priors is utilized, to embed object instances in a unique space and learn from a large collection of CAD models. We further utilize differentiable optimization of implicit shape and appearance representation to iteratively improve shape, pose, size and appearance jointly in an analysis-by-synthesis fashion. To alleviate the sampling inefficiency inherent in signed distance field shape representations, we propose a novel octree-based point sampling which leads to significant time and memory improvements as well as increased reconstruction quality.

Our contributions are summarized as follows:

- **A generalizable, disentangled shape and appearance space** coupled with **an efficient octree-based differentiable optimization procedure** which allows us to identify and reconstruct novel object instances without access to their ground truth meshes.
- **An object-centric scene understanding pipeline** relying on learned joint appearance and implicit differentiable shape priors which achieves state-of-the-art reconstruction and pose estimation results on benchmark datasets.
- Our proposed approach significantly outperforms all baselines for 6D pose and size estimation on NOCS benchmark, showing over over 8% absolute improvement in mAP for 6D pose at $10°$ $10\,\text{cm}$.

2   Related Work

In essence, our proposed method infers 3D shape along with predicting the 3D appearance and 6D pose and sizes of multiple object instances to perform object-centric scene reconstruction from a single-view RGB-D, so it relates to multiple areas in 3D scene reconstruction, object understanding and pose estimation.

Neural Implicit Representations: 3D shape and appearance reconstructions has recently seen a new prominent direction to use neural nets as scalar fields approximators instead of ConvNets. The first works of this class are notably DeepSDF [36], Occ-Net [29], and IM-Net [4]. These works use *coordinate based representation* to output either an occupancy estimate or a continuous SDF value, encoding the object's surface given a 3D coordinate. Improving this direction further, MeshSDF [42], NGLOD [48] and Texture fields [34] employed end-to-end differentiable mesh representation, efficient octree representation and implicitly representing high frequency textures respectively. In our pipeline, we build a differentiable database of shape and appearance priors based on the latest

advances in neural implicit representations. Our pipeline is end-to-end differentiable and abstains from the expensive Marching Cubes computation at every optimization step. Our database stores not only the geometries of the objects in the form of signed distance fields, but also their appearance in the form of texture fields, allowing us to model multiple categories through a single network while also considering test-time optimization of shape, pose and appearance.

Object-centric Reconstruction: 3D object reconstruction from a single-view observation has seen great progress to output pointclouds, voxels or meshes [4, 29, 36]. Similarly, scene representation has been extended to include appearance. SRN [45], DVR [33] learn from multi-view images by employing ray-marching routine and differentiable rendering respectively. Recently, NeRF [30] propose to regress density and color along a ray and perform volumetric rendering to obtain true color value. Most NeRF-based methods [27, 32, 35] overfit to a single scene, do not promise generalizability and requires dense viewpoint annotations. Our pipeline, on the other hand, is capable of reconstructing shapes, appearances and inferring the 6D pose and sizes of objects never seen during training from a single-view RGB-D and is also not limited to viewpoints seen during training.

6DoF Object Pose and size estimation works use direct pose regression [16, 21, 53, 57], template matching [22, 46, 49] and establishing correspondences [12, 15, 37, 43, 54]. However, most works focus only on pose estimation and do not deal with shape and appearance retrieval and their connection to 6D pose. We instead propose a differentiable pipeline to improve the initially regressed pose, along with the shape and appearance, using our novel octree-base differentiable test-time optimization.

3   Method

ShAPO is an end-to-end learning-based holistic object-centric scene understanding method that infers 3D shape along with the 6D pose, size and appearance of multiple unknown objects in an RGB-D observation. ShAPO tackles the detection, localization and reconstruction of all unknown object instances in 3D space. Such a goal is made possible by three components: 1) A single-shot detection and 3D prediction module that detects multiple objects based on their center points in the 2D spatial grid and recovers their complete 3D shapes, 6D pose and sizes along with appearance from partial observations. 2) An implicit joint differentiable database of shape and appearances priors which is used to embed objects in a unique space and represent shapes as signed distance fields (SDF) and appearance as continuous texture fields (TF). 3) A 2D/3D refinement method utilizing an octree-based coarse-to-fine differentiable optimization to improve shape, pose, size and appearance predictions iteratively.

Formally, given a single-view RGB-D observation ($I \in \mathbb{R}^{h_o \times w_o \times 3}$, $D \in \mathbb{R}^{h_o \times w_o}$) as input, ShAPO infers the complete 3D information of multiple objects including the shape (represented as a zero-level set of an implicit signed distance field, $\boldsymbol{SDF}$), 6D pose and scales ($\tilde{\mathcal{P}} \in SE(3)$, $\hat{s} \in \mathbb{R}^3$) and appearance (represented as a continuous texture field, $\boldsymbol{TF}$). To achieve the above goal, we employ an end-to-end trainable method. As shown in Figure 2, we first formulate
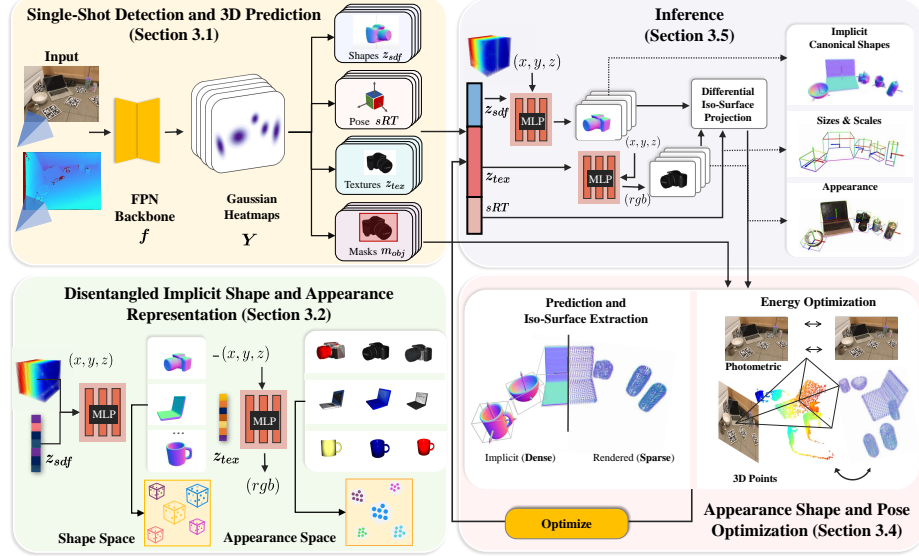
Fig. 2: **ShAPO Method:** Given a single-view RGB-D observation, our method jointly predicts shape, pose, size and appearance codes along with the masks of each object instances in a single-shot manner. Leveraging a novel octree-based differentiable optimization procedure, we further optimize for shape, pose, size and appearance jointly in an analysis by synthesis manner.

object detection as a spatial per-pixel point detection [16, 25, 61]. A backbone comprising of feature pyramid networks [10] is employed with a specialized head to predict object instances as heatmaps in a per-pixel manner. Second, joint shape, pose, size and appearance codes along with instance masks are predicted for detected object centers using specialized heads (Section 3.1). Our combined shape and appearance implicit differentiable database of priors is described in Section 3.2 and the corresponding zero iso-surface based differentiable rendering is detailed in Section 3.4. Lastly, 3D shapes along with their appearance, coupled with 6D pose and sizes of novel objects are jointly optimized during inference to predict accurate shapes, poses and sizes of novel object instances (Section 3.5).

3.1   Single-Shot Detection and 3D prediction

ShAPO represents object instances along with their complete 3D information including shape, pose, appearance and size codes along with corresponding 2D masks through their 2D location in the spatial RGB image, following [7,16,25,61]. Given an RGB-D observation ($I \in \mathbb{R}^{h_o \times w_o \times 3}$, $D \in \mathbb{R}^{h_o \times w_o}$), ShAPO predicts object-centric heat maps $\hat{Y} \in [0,1]^{\frac{h_o}{R} \times \frac{w_o}{R} \times 1}$ where each detected point $(\hat{x}_i, \hat{y}_i)$ denotes the local maxima in the heatmap ($\hat{Y}$). Here R denotes the heatmap down-sampling factor, and is set to 8 in all our experiments. To predict these heatmaps, a feature backbone based on feature pyramid networks (FPN) [23] is utilized along with a specialized heatmap prediction head. During training, we

find the target heatmaps by splatting the ground truth center points (x_i, y_i) using the gaussian kernel $\mathcal{N}(x_i, y_i, \sigma_i)$ where σ_i is relative to the spatial extent of each object (as defined by the corresponding ground truth bounding box annotation). The network is trained to predict ground-truth heatmaps (Y) by minimizing MSE loss over all pixels (x, y) in the heatmap, $\mathcal{L}_{\text{inst}} = \sum_{xyg} \left(\hat{Y} - Y \right)^2$. The network also predicts object instance masks $(\hat{M})$ using a specialized head (f_{θ_m}) to output $\hat{M} \in \mathbb{R}^{h_o \times w_o}$, similar to the semantic segmentation head described in [23]. Note that it is crucial for the network to predict masks for accurate downstream optimization (see Section 3.4).

3.2 Joint Implicit Shape, Pose and Appearance Prediction

Once the network detects objects, it then predicts their complete 3D information (i.e. 3D shape, 6D pose and size along with the 3D appearance) all in a single-forward pass using specialized heads $(f_{\theta_{sdf}}, f_{\theta_P}$ and $f_{\theta_{tex}})$ with outputs $(Y_{sdf} \in \mathbb{R}^{\frac{h_o}{R} \times \frac{w_o}{R} \times 64}, Y_P \in \mathbb{R}^{\frac{h_o}{R} \times \frac{w_o}{R} \times 13}$ and $Y_{tex} \in \mathbb{R}^{\frac{h_o}{R} \times \frac{w_o}{R} \times 64}$ respectively). During training, the task-specific heads output shape code z_{sdf}, 6D pose $\tilde{\mathcal{P}}$, scale $\hat{s}$ and appearance z_{tex} information at each pixel in the downsampled map $(\frac{h_o}{R} \times \frac{w_o}{R})$. For each object's Pose $(\tilde{\mathcal{P}})$ with respect to the camera coordinate frame, we regress a 3D rotation $\hat{\mathcal{R}} \in SO(3)$, a 3D translation $\hat{t} \in \mathbb{R}^3$ and 1D scales $\hat{s}$ (totaling thirteen numbers). These parameters are used to transform the object shape from canonical frame to the 3D world. We select a 9D rotation $\hat{\mathcal{R}} \in SO(3)$ representation since the neural network can better fit a continuous representation and to avoid discontinuities with lower rotation dimensions, as noted in [62]. Furthermore, we employ a rotation mapping function following [40] to handle ambiguities caused by rotational symmetries. The rotation mapping function is used only for for symmetric objects *(bottle, bowl, and can)* in our database during training and it maps ambiguous ground-truth rotations to a single canonical rotation by normalizing the pose rotation. Note that during training, ground-truth shape codes z_{sdf} and appearance codes z_{tex} for each object are obtained from our novel implicit textured differentiable representation (further described in Section 3.3).

During training, we jointly optimize for shape, pose, appearance and mask prediction. Specifically, we minimize the masked L_1 loss for shape, pose and appearance prediction, denoted as $\mathcal{L}_{\text{sdf}}, \mathcal{L}_{\text{tex}}, \mathcal{L}_{\text{P}}$ and a pixel-wise cross-entropy loss for mask prediction $\mathcal{L}_{\text{M}} = \sum_{i=1}^{h_o \cdot w_o} - \log \hat{M}_i \left(M_i^{\text{gt}} \right)$ where M_i^{gt} denotes the ground truth category label for each pixel.

During training, we minimize a combination of these losses as follows:

$$\mathcal{L} = \lambda_{inst}\mathcal{L}_{inst} + \lambda_{sdf}\mathcal{L}_{sdf} + \lambda_{tex}\mathcal{L}_{tex} + \lambda_M\mathcal{L}_M + \lambda_P\mathcal{L}_P \qquad (1)$$

where λ is a weighting co-efficient with values determined empirically as $lambda_{inst} = 100$ and $\lambda_{sdf} = \lambda_{tex} = \lambda_P = 1.0$. Note that for shape, appearance and pose predictions, we enforce the L_1 loss based on the probability estimates of the Gaussian heatmaps (Y) i.e. the loss is only applied where Y has a score

greater than 0.3 to prevent ambiguity in the space where no object exists. We now describe the shape and appearance representation utilized by our method.

3.3   Implicit textured differentiable database of priors

We propose a novel joint implicit textured representation to learn from a large variety of CAD models and embed objects in a concise space. This novel representation (as shown in Figure 2 and Figure 3) is also used as a strong inductive prior to efficiently optimize the shape and appearance along with the pose and size of objects in a differentiable manner (Section 3.4). In our implicit textured databse of shape and appearance priors, each object shape is represented as a Signed Distance Field (SDF) where a neural network learns a signed distance function $G(\boldsymbol{x}, \boldsymbol{z}_{sdf}) = s : z_{sdf} \in \mathbb{R}^{64}, s \in \mathbb{R}$ for every 3D point $x \in \mathbb{R}^3$ and the appearance is represented as Texture Fields ($t_\theta : \mathbb{R}^3 \to \mathbb{R}^3$) which maps a 3D point $x \in \mathbb{R}^3$ to an RGB value $c \in \mathbb{R}^3$. Since the mapping between coordinates and colors is ambiguous without shape information, we propose to learn a texture field only at the predicted shape i.e. $t_\theta(\boldsymbol{x}, \boldsymbol{z}_{sdf}, \boldsymbol{z}_{tex}) = c, \boldsymbol{z}_{tex} \in \mathbb{R}^{64}$. The SDF function ($G$) implicitly defines the surface of each object shape by the zero-level set $G(.) = 0$. To learn a shape-code ($\boldsymbol{z}_{sdf}$) and texture code ($\boldsymbol{z}_{tex}$) for each object instance, we design a single MLP (multi-layer perceptron) each for shape (to reason about the different geometries in the database) and texture (to predict color information given shape and texture codes). Through conditioning the MLP output on the latent vector, we allow modeling multiple geometries and appearances using a single network each for shape and appearance. Each MLP is trained separately using the supervised ground-truth reconstruction loss L_{SDF} and L_{RGB} as follows:

$$L_{SDF} = |\text{clamp}\,(G(\boldsymbol{x}, \boldsymbol{z}_{sdf}), \delta) - \text{clamp}\,(\boldsymbol{s}_{gt}, \delta)| + L_{contrastive}(\boldsymbol{z}_{sdf}) \qquad (2)$$

$$L_{RGB} = \sum_{n=1}^{N} \|\boldsymbol{c}_{gt} - t_\theta(\boldsymbol{x}, \boldsymbol{z}_{sdf}, \boldsymbol{z}_{tex})\|_2^2 \qquad (3)$$

where L_{SDF} is a combination of a clipped L_1 loss between ground-truth signed-distance values $\boldsymbol{s}_{gt}$ and predicted SDF $G(\boldsymbol{x}, \boldsymbol{z}_{sdf})$ and a contrastive loss $L_{\text{contrastive}} = [m_{\text{pos}} - s_p]_+ + [s_n - m_{\text{neg}}]_+$. As shown by the t-SNE embeddings [28] (Figure 3) for the latent shape-code ($\boldsymbol{z}_{sdf}$), the contrastive loss helps with disentangling the shape space nicely and leads to better downstream regression in the single-shot model (Section 3.2). Once we train the implicit shape auto-decoder, we use the learned shape space $\boldsymbol{z}_{sdf}$ to minimize the color loss L_{RGB} which is defined as an MSE loss between predicted color at the surface $t_\theta(\boldsymbol{x}, \boldsymbol{z}_{sdf}, \boldsymbol{z}_{tex})$ and ground-truth color $\boldsymbol{c}_{gt}$.

We use 3D textured models from the CAD model repository Shapenet [1] to learn our databse of shape and texture priors. Once trained, the MLP networks for both shape and appearance find a disentangled space for color and geometry while keeping semantically-similar objects together (Figure 3) and provides us with strong priors to be used for 2D and 3D optimization (described in Section 3.4).
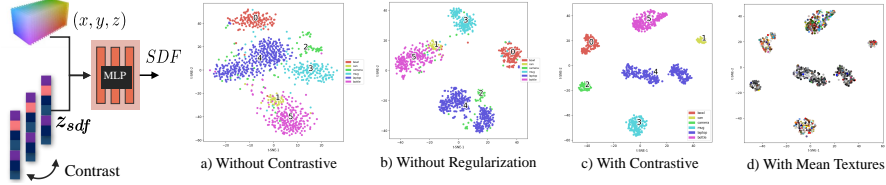
Fig. 3: Deep Implicit shape space for NOCS object Categories

3.4 Differentiable Optimization

A key component of ShAPO is the optimization scheme allowing to refine initial object predictions with respect to the pose, scale, shape, and appearance. Inspired by sdflabel [59], we develop a new differentiable and fast optimization method. Instead of using mesh-based representations, we rely entirely on implicit surfaces, which not only helps us avoid common connectivity and intersection problems, but also provides us full control over sampling density.

Surface Projection Given input grid points x_i and estimated SDF values s_i, we aim to find a differentiable transformation to extract the object surface encoded in $\mathbf{z}_{sdf}$. A trivial solution would be to simply threshold points with SDF values more than a certain threshold. However, this procedure is not differentiable with respect to the input latent vector $\mathbf{z}_{sdf}$. Instead, we utilize the fact that deriving an SDF value s_i with respect to its input coordinate x_i yields a normal at this point, which can be computed in a single backward pass:

$$n_i = \frac{\partial G(x_i; \mathbf{z}_{sdf})}{\partial x_i}. \quad \text{and} \quad p_i = x_i - \frac{\partial G(x_i; \mathbf{z}_{sdf})}{\partial x_i} G(x_i; \mathbf{z}_{sdf}). \tag{4}$$

Octree-based Point Sampling The brute force solution to recover shapes from a learned SDF representation can be obtained by estimating SDF values for a large collection of grid points similar to the procedure used in [59]. To obtain clean surface projections one would then disregard all points x_i outside a narrow band ($|s_i| > 0.03$) of the surface. However, this procedure is extremely inefficient both memory- and compute-wise — For a grid size of $60^3 = 216000$ points, only around 1600 surface points are extracted (accounting to 0.7% of the total number of points). We propose an Octree-based procedure to efficiently extract points. We define a coarse voxel grid and estimate SDF values for each of the points using our trained SDF network. We then disregard voxels whose
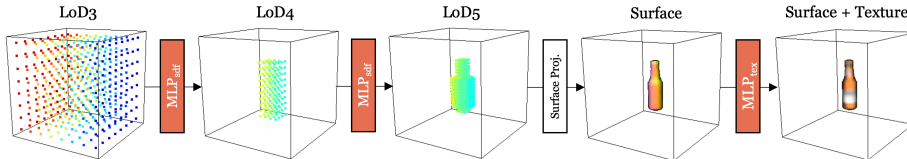

Fig. 4: Octree-based object extraction.

Algorithm 1: Octree-based implicit surface extraction

Input: $\mathbf{x} \in \mathbb{R}^3$ grid points, $\mathbf{l} \in \mathbb{L}$ grid levels, $\mathbf{z}_{sdf}$ and $\mathbf{z}_{tex} \in \mathbb{R}^{64}$ latent vectors
Output: $\mathbf{pcd} \in \mathbb{R}^3$ surface points, $\mathbf{nrm} \in \mathbb{N}^3$ normals, $\mathbf{col} \in \mathbb{C}^3$ colors

```
/* Extract object grid (no grad)                                    */
```
1 for $l \in \{1, \ldots, N_{LoD}\}$ **do**
2 | $\mathbf{sdf} \leftarrow G(\mathbf{x}_l, \mathbf{z}_{sdf})$;   `// regress sdf values`
3 | $\mathbf{occ} \leftarrow \mathbf{sdf} < getCellSize(l)$;   `// estimate cell occupancy`
4 | $\mathbf{x}_{l_{occ}} \leftarrow \mathbf{x}_l[occ]$;   `// remove unoccupied cells`
5 | $\mathbf{x}_{l+1} \leftarrow subdivide(\mathbf{x}_{l_{occ}})$;   `// subdivide cells to go to next LoD`
6 end
```
/* Extract object shape and appearance                             */
```
7 nrm $\leftarrow$ backprop($\mathbf{sdf}$) ;   `// analytically estimate surface normals`
8 pcd $\leftarrow \mathbf{x} - \mathbf{nrm} * \mathbf{sdf}$;   `// project points onto the surface`
9 col $\leftarrow t_\theta(\mathbf{pcd}, \mathbf{z}_{sdf}, \mathbf{z}_{tex})$;   `// regress surface texture`
10 return *pcd, nrm, col*

SDF values are larger than the voxel grid size for this resolution level. The remaining voxels are subdivided each generating eight new voxels. We repeat this procedure until the desired resolution level is reached. In our implementation, we start from level of detail (LoD) 3 and traverse up to LoD 6 depending on the desired resolution level. Finally, when points are extracted we estimate their SDF values and normals and project them them onto the object surface. The pseudo-code implementation of the Octree-based surface extraction is provided in Alg. 1 with the visualization shown in Fig. 4.

3.5   Inference:

During inference, we first perform predictions using our single-shot model. Object detection is performed using peak detection [61] on the outputs of predicted heatmaps $(\hat{Y})$. Each detected center point $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ corresponds to maxima in the heatmap output $(\hat{Y})$. Second, we sample shape, pose and appearance codes of each object from the output of task-specific heads at the detected center location (x_i, y_i) via $\boldsymbol{z}_{sdf} = Y_{sdf}(\boldsymbol{x}_i, \boldsymbol{y}_i)$, $\boldsymbol{z}_{tex} = Y_{tex}(\boldsymbol{x}_i, \boldsymbol{y}_i)$ and $\tilde{\mathcal{P}} = Y_P(\boldsymbol{x}_i, \boldsymbol{y}_i)$.

We utilize the predicted shape, pose, size and appearance codes as an initial estimate to further refine through our differentiable optimization pipeline. Our optimizer takes as input the predicted implicit shapes in the canonical frame of reference along with the masks predictions $(\hat{M})$, color codes $(\boldsymbol{z}_{tex})$ and extracted 3×3 rotation $\hat{\mathcal{R}}_i^p$, 3D translation vector $\hat{t}_i^p$ and 1D scales $\hat{s}_i^p$ from recovered Pose $\tilde{\mathcal{P}}$. Although a variety of related works consider mean class predictions as initial priors, we mainly utilize the regressed outputs of shape, appearance and pose for the optimization pipeline since the initial estimates are very robust (see Table 1). We utilize the predicted SDF to recover the complete surface of each object, in a coarse-to-fine manner, using the proposed differentiable zero-isosurface projection (Section 3.4). After fixing the decoder (G) parameters, we optimize the feature vector $\boldsymbol{z}_{sdf}$ by estimating the nearest neighbour between the pre-

Table 1: **Quantitative comparison of 6D pose estimation and 3D object detection on NOCS** [54]: Comparison with strong baselines. Best results are highlighted in **bold**. ∗ denotes the method does not report IOU metrics since size and scale is not evaluated. We report metrics using nocs-level class predictions for a fair comparison with all baselines.

Method	CAMERA25						REAL275					
	IOU25	IOU50	5°5 cm	5°10 cm	10°5 cm	10°10 cm	IOU25	IOU50	5°5 cm	5°10 cm	10°5 cm	10°10 cm
1 NOCS [54]	91.1	83.9	40.9	38.6	64.6	65.1	84.8	78.0	10.0	9.8	25.2	25.8
2 Synthesis* [3]	-	-	-	-	-	-	-	-	0.9	1.4	2.4	5.5
3 Metric Scale [26]	93.8	90.7	20.2	28.2	55.4	58.9	81.6	68.1	5.3	5.5	24.7	26.5
4 ShapePrior [51]	81.6	72.4	59.0	59.6	81.0	81.3	81.2	77.3	21.4	21.4	54.1	54.1
5 CASS [2]	-	-	-	-	-	-	84.2	77.7	23.5	23.8	58.0	58.3
6 CenterSnap [16]	93.2	92.3	63.0	69.5	79.5	87.9	83.5	80.2	27.2	29.2	58.8	64.4
7 CenterSnap-R [16]	93.2	92.5	66.2	71.7	81.3	87.9	83.5	**80.2**	29.1	31.6	64.3	70.9
8 ShAPO (Ours)	**94.5**	**93.5**	**66.6**	**75.9**	**81.9**	**89.2**	**85.3**	79.0	**48.8**	**57.0**	**66.8**	**78.0**

dicted projected pointclouds and masked pointclouds obtained from depth map and predicted masks ($\hat{M}$) of each object. In essence, a shape code $\boldsymbol{z}_{sdf}$ is refined using the Maximum-a-Posterior (MAP) estimation as follows:

$$\boldsymbol{z}_{sdf} = \arg\min_{\boldsymbol{z}}(\mathcal{L}\left(D(G(\boldsymbol{z}, \boldsymbol{x})), P_d\right) \tag{5}$$

where D() denotes the differentiable iso-surface projection described in Section 3.4 and P_d denotes the pointclouds obtained from masked depth maps. We further optimize the RGB component similarly by optimizing the difference in colors between the masked image color values (C_d) and colors obtained using the regressed color codes decoded by the texture field (t_θ) $\boldsymbol{z}_{tex} = \arg\min_{\boldsymbol{z}}(\mathcal{L}\left(D(t_\theta(\boldsymbol{z}, \boldsymbol{x})), C_d\right)$. We further allow t_θ weights to change to allow for a finer level of reconstruction. For qualitative results please consult supplementary material.

4   Experiments

In this section, we empirically validate the peformance of our method. In essence, our goal is to answer these questions: 1) How well does ShAPO recover pose and sizes of novel objects. 2) How well does ShAPO perform in terms of reconstructing geometry and appearance of multiple objects from a single-view RGB-D observation? 3) How well does does our differentiable iterative improvement and multi-level optimization impact shape, appearance, pose and size?

Datasets We utilize the **NOCS** [54] dataset for both shape reconstruction and category-level 6D pose and size estimation evaluation. For training, we utilize the CAMERA dataset which comprises 300K synthetic images, of which 25K are hold-out for evaluation. The training dataset includes 1085 object object models from 6 different categories - *bottle, bowl, camera, can, laptop and mug* whereas

Table 2: **Quantitative comparison of 3D shape reconstruction on NOCS** [54]: Evaluated with **CD** metric (10^{-2}). Lower is better.

Method	CAMERA25							REAL275						
	Bottle	Bowl	Camera	Can	Laptop	Mug	Mean	Bottle	Bowl	Camera	Can	Laptop	Mug	Mean
1 Reconstruction [51]	0.18	0.16	0.40	0.097	0.20	0.14	0.20	0.34	0.12	0.89	0.15	0.29	0.10	0.32
2 ShapePrior [51]	0.34	0.22	0.90	0.22	0.33	0.21	0.37	0.50	0.12	0.99	0.24	0.71	0.097	0.44
3 CenterSnap	0.11	0.10	0.29	0.13	0.07	0.12	0.14	0.13	0.10	0.43	0.09	0.07	0.06	0.15
3 **ShAPO (Ours)**	0.14	0.08	0.2	0.14	0.07	0.11	0.16	0.1	0.08	0.4	0.07	0.08	0.06	0.13

the evaluation dataset includes 184 different models. The REAL dataset train-set comprises of 7 scenes with 4300 images, and test-set comprises of 6 scenes with 2750 real-world images.

Metrics We evaluate the the performance of 3D object detection and 6D pose estimation independently following [54]. We use the following key metrics: 1) Average-precision for different IOU-overlap thresholds (**IOU25** and **IOU50**). 2) Average precision for which the error is less than $n°$ for rotation and m cm for translation (**5°**5 cm, **5°**10 cm and **10°**10 cm). We use Chamfer distance (CD) for shape reconstruction following [58].

4.1   Implementation Details

ShAPO is sequentially trained first on the CAMERA set with minimal fine-tuning on Real training set. For SDF, we use an MLP with 8 layers and hidden size of 512. For color, we utilize a Siren MLP [44] as it can fit higher frequencies better. We train the SDF and Color MLPs on all categories for 2000 epochs. We use Pytorch [38] for our models and training pipeline implementation. For optimization, we use an adaptive learning rate which varies with the obtained masks of each object since we believe masks capture the confidence of heatmap prediction during detection. We optimize each object for 200 iterations.
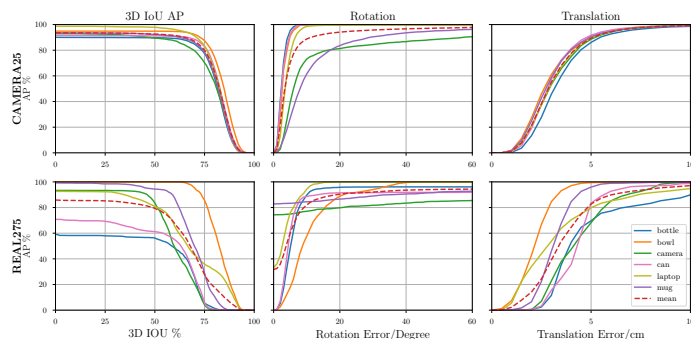

Fig. 5: Average precision of ShAPO for various IOU, rotation and translation thresholds on NOCS CAMERA25 and REAL275 dataset.

4.2 NOCS 6D Pose and Size Estimation Benchmark

NOCS Baselines: We compare eight model variants to show effectiveness of our method: (1) **NOCS** [54]: Regresses NOCS map and predicts poses using similarity transform with depth maps. We report the best pose estimation configuration in NOCS (i.e. 32-bin classification) (2) **Shape Prior** [51]: Estimates shape deformation for inter-class variation. (3) **CASS** [2]: Regress the pose and size with first detecting 2D bounding boxes. (4) **Metric-Scale** [26]: Estimates the object center and metric scale (5) **CenterSnap** [16]**:** Single-shot approach to predict pose and size (6) **CenterSnap-R** [16]**:** ICP optimization done with the predicted point-cloud based shape. Following [16], we do not compare against 6D pose tracking baselines such as [52, 56] which need pose initializations and are not end-to-end detection based (i.e. they do not report mAP metrics)

Comparison with strong baselines on NOCS: Table 1 and Figure 5 shows the result of our proposed ShAPO method. ShAPO consistently outperforms all the baseline methods on 6D pose and size estimation and 3D object detection. Specifically, ShAPO method shows superior generalization on the REAL test-set by achieving a mAP of 85.3% for 3D IOU at 0.25, 57.0% for 6D pose at 5°10 cm and 78.0% for 6D pose at 10°10 cm, hence demonstrating an absolute improvement of 1.8%, 25.4% and 7.1% over the best-performing baseline on the Real dataset. Our method also achieves better generalization on the never-seen CAMERA evaluation set. We achieve a mAP of 94.5% for 3D IOU at 0.25, 75.9% for 6D pose at 5°10 cm and 89.2% for 6D pose at 10°10 cm, demonstrating an absolute improvement of 1.3%, 4.2% and 1.3% over the best-performing baseline.

4.3 Generalizable Implicit Object Representation

In this experiment, we evaluate the effectiveness of our generalizable implicit object representation as well as the efficiency of our octree-based differentiable optimization. To do that, we isolate our implicit representation from the detection pipeline and consider the 3D reconstruction / identification task - given a novel object from NOCS test split, we optimize our representation for 200 iterations, while keeping f_{sdf} and f_{rgb} weights frozen, to find the best fitting model in terms of both shape and texture. We initialize the optimization using the average latent feature per class. The standard Adam solver with the learning rate of 0.001 for both for both shape and appearance losses (L2 norms) is used with weight factors 1 and 0.3 respectively. We use three different octree resolution levels - from LoD5 to LoD7. Additionally, we show three different resolution levels for the standard ordinary grid sampling [59] (40, 50, 60).

Table 3 summarizes the results of the ablation by comparing different modalities with respect to the average point sampling (input vs output) and time efficiency, average GPU memory consumption, as well as reconstruction for shape (Chamfer distance) and texture (PSNR). One can see that our representation is significantly more efficient than the ordinary grid representation with respect to

Table 3: **Generalizable Implicit Representation Ablation**: We evaluate the efficiency (point sampling/time(s)/memory(MB)) and generalization (shape(CD) and texture(PSNR) reconstruction) capabilities of our implicit object representation as well as its sampling efficiency for different levels of detail (LoDs) and compare it to the ordinary grid sampling. All ablations were executed on NVIDIA RTX A6000 GPU.

Grid type	Resolution	Point Sampling		Efficiency (per object)		Reconstruction	
		Input	Output	Time (s)	Memory (MB)	Shape (CD)	Texture (PSNR)
Ordinary	40	64000	412	10.96	3994	0.30	10.08
	50	125000	835	18.78	5570	0.19	12.83
	60	216000	1400	30.51	7850	0.33	**19.52**
OctGrid	LoD5	1521	704	**5.53**	**2376**	0.19	9.27
	LoD6	5192	3228	6.88	2880	**0.18**	13.63
	LoD7	20246	13023	12.29	5848	0.24	16.14

all metrics. While LoD7 provides best overall results, we use LoD6 for our experiments since it results in the optimal speed/memory/reconstruction trade-off. We show an example optimization procedure in Fig. 6.

4.4   NOCS Object Reconstruction Benchmark

To quantitatively analyze the reconstruction accuracy, we measure the Chamfer distance (CD) between our reconstructed pointclouds and the ground-truth CAD model in NOCS. Our results are reported in Table 2. Our results show consistently lower CD metrics for all class categories which shows superior reconstruction performance on novel object instances. We report a lower mean Chamfer distance of 0.14 on CAMERA25 and 0.15 on REAL275 compared to 0.20 and 0.32 reported by the competitive baseline [51].

5   Qualitative Results

We qualitatively analyze the performance of our method **ShAPO** on the NOCS Real275 dataset [54]. As shown in Fig. 7, our method reconstructs accurate 3D shapes and appearances of multiple novel objects along with estimating the 6D pose and sizes without requiring 3D CAD models of these novel instances (Results shown on 4 different real-world scenes containing novel object instances using different camera-view points i.e. orange and green backgrounds). For further qualitative results, please consult supplementary materials.
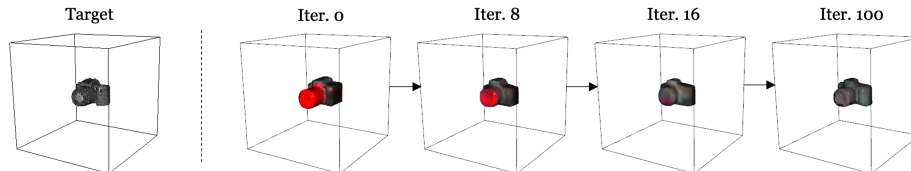

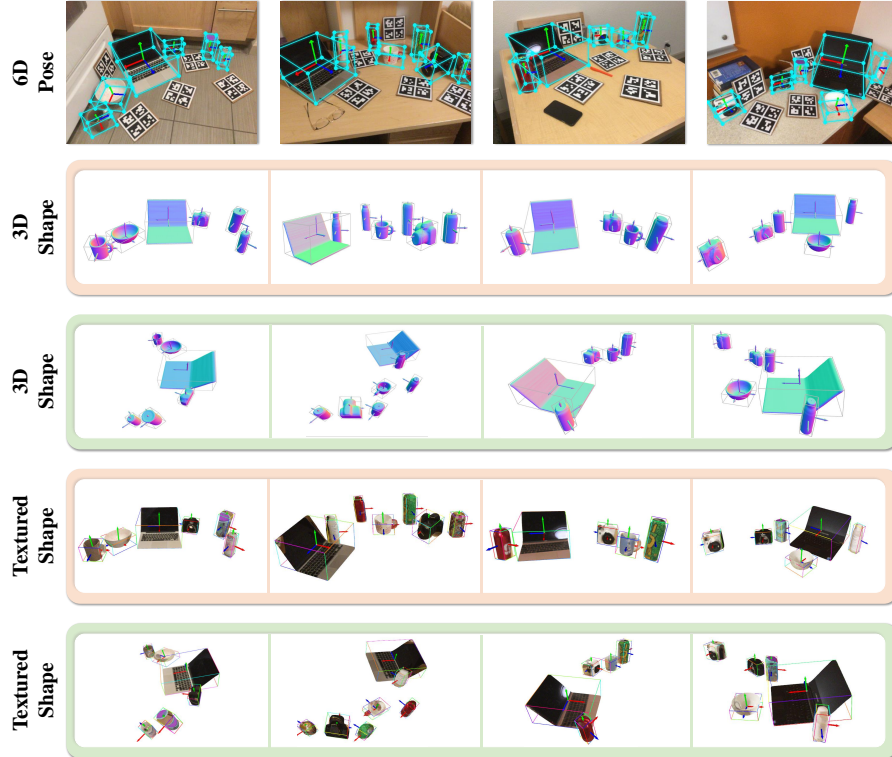
Fig. 6: Shape/Appearance optimization.

Fig. 7: **ShAPO Qualitative 6D pose estimation and 3D Shape Reconstructions including Appearance:** Given a single-view RGB-D observation, our method reconstructs accurate 3D shapes along with appearances and estimates the 6D pose and sizes of multiple objects in the scene. Here, reconstructions are shown with different camera view-points i.e. orange and green backgrounds.

6   Conclusion

In this paper we proposed ShAPO , an end-to-end method for joint multi-object detection, 3D textured reconstruction, 6D object pose and size estimation. Our method detects and reconstructs novel objects without having access to their ground truth 3D meshes. To facilitate this, we proposed a novel, generalizable shape and appearance space that yields accurate textured 3D reconstructions of objects in the wild. To alleviate sampling inefficiencies leading to increased time and memory requirements, we proposed a novel octree-based differentiable optimization procedure that is significantly more efficient than alternative grid based representations. For future work we will explore how the proposed method can be used to build object databases in new environments, to alleviate the cost and time required to construct high-quality 3D textured assets. A second avenue of future work consists of extensions to multi-view settings and integrating our work with SLAM pipelines for joint camera motion and object pose, shape and texture estimation in static and dynamic scenes.

References

1. Chang, A.X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., et al.: ShapeNet: An information-rich 3d model repository. arXiv preprint arXiv:1512.03012 (2015)
2. Chen, D., Li, J., Wang, Z., Xu, K.: Learning canonical shape space for category-level 6d object pose and size estimation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11973–11982 (2020)
3. Chen, X., Dong, Z., Song, J., Geiger, A., Hilliges, O.: Category level object pose estimation via neural analysis-by-synthesis. In: European Conference on Computer Vision. pp. 139–156. Springer (2020)
4. Chen, Z., Zhang, H.: Learning implicit fields for generative shape modeling. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5939–5948 (2019)
5. Choy, C.B., Xu, D., Gwak, J., Chen, K., Savarese, S.: 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In: European conference on computer vision. pp. 628–644. Springer (2016)
6. Cifuentes, C.G., Issac, J., Wüthrich, M., Schaal, S., Bohg, J.: Probabilistic articulated real-time tracking for robot manipulation. IEEE Robotics and Automation Letters **2**(2), 577–584 (2016)
7. Duan, K., Bai, S., Xie, L., Qi, H., Huang, Q., Tian, Q.: CenterNet: Keypoint triplets for object detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6569–6578 (2019)
8. Fan, H., Su, H., Guibas, L.J.: A point set generation network for 3d object reconstruction from a single image. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 605–613 (2017)
9. Ferrari, C., Canny, J.F.: Planning Optimal Grasps. In: ICRA. vol. 3, p. 6 (1992)
10. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: Computer Vision and Pattern Recognition (2014)
11. Gkioxari, G., Malik, J., Johnson, J.: Mesh R-CNN. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9785–9795 (2019)
12. Goodwin, W., Vaze, S., Havoutis, I., Posner, I.: Zero-shot category-level object pose estimation. arXiv preprint (2022)
13. Groueix, T., Fisher, M., Kim, V.G., Russell, B., Aubry, M.: AtlasNet: A Papier-Mâché Approach to Learning 3D Surface Generation. In: Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR) (2018)
14. He, K., Gkioxari, G., Dollár, P., Girshick, R.B.: Mask R-CNN. In: ICCV (2017)
15. Hodan, T., Barath, D., Matas, J.: Epos: estimating 6d pose of objects with symmetries. In: CVPR (2020)
16. Irshad, M.Z., Kollar, T., Laskey, M., Stone, K., Kira, Z.: Centersnap: Single-shot multi-object 3d shape reconstruction and categorical 6d pose and size estimation. In: IEEE International Conference on Robotics and Automation (ICRA) (2022), https://arxiv.org/abs/2203.01929
17. Irshad, M.Z., Ma, C.Y., Kira, Z.: Hierarchical cross-modal agent for robotics vision-and-language navigation. In: 2021 IEEE International Conference on Robotics and Automation (ICRA). pp. 13238–13246 (2021). https://doi.org/10.1109/ICRA48506.2021.9561806

18. Irshad, M.Z., Mithun, N.C., Seymour, Z., Chiu, H.P., Samarasekera, S., Kumar, R.: Sasra: Semantically-aware spatio-temporal reasoning agent for vision-and-language navigation in continuous environments (2022), https://arxiv.org/abs/2108.11945

19. Jiang, Z., Zhu, Y., Svetlik, M., Fang, K., Zhu, Y.: Synergies between affordance and geometry: 6-Dof grasp detection via implicit representations. Robotics: science and systems (2021)

20. Kato, H., Ushiku, Y., Harada, T.: Neural 3d mesh renderer. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3907–3916 (2018)

21. Kehl, W., Manhardt, F., Tombari, F., Ilic, S., Navab, N.: Ssd-6d: Making rgb-based 3d detection and 6d pose estimation great again. In: ICCV (2017)

22. Kehl, W., Milletari, F., Tombari, F., Ilic, S., Navab, N.: Deep learning of local rgb-d patches for 3d object detection and 6d pose estimation. In: European conference on computer vision. pp. 205–220. Springer (2016)

23. Kirillov, A., Girshick, R., He, K., Dollár, P.: Panoptic feature pyramid networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6399–6408 (2019)

24. Kuo, W., Angelova, A., Lin, T.Y., Dai, A.: Mask2CAD: 3D shape prediction by learning to segment and retrieve. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16. pp. 260–277. Springer (2020)

25. Laskey, M., Thananjeyan, B., Stone, K., Kollar, T., Tjersland, M.: SimNet: Enabling robust unknown object manipulation from pure synthetic data via stereo. In: 5th Annual Conference on Robot Learning (2021)

26. Lee, T., Lee, B.U., Kim, M., Kweon, I.S.: Category-level metric scale object shape and pose estimation. IEEE Robotics and Automation Letters $\mathbf{6}$(4), 8575–8582 (2021)

27. Li, Z., Niklaus, S., Snavely, N., Wang, O.: Neural scene flow fields for space-time view synthesis of dynamic scenes. In: CVPR (2021)

28. Van der Maaten, L., Hinton, G.: Visualizing data using t-SNE. Journal of machine learning research $\mathbf{9}$(11) (2008)

29. Mescheder, L., Oechsle, M., Niemeyer, M., Nowozin, S., Geiger, A.: Occupancy networks: Learning 3d reconstruction in function space. In: CVPR (2019)

30. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. In: ECCV (2020)

31. Nie, Y., Han, X., Guo, S., Zheng, Y., Chang, J., Zhang, J.J.: Total3dunderstanding: Joint layout, object pose and mesh reconstruction for indoor scenes from a single image. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2020)

32. Niemeyer, M., Geiger, A.: Giraffe: Representing scenes as compositional generative neural feature fields. In: Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR) (2021)

33. Niemeyer, M., Mescheder, L., Oechsle, M., Geiger, A.: Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In: CVPR (2020)

34. Oechsle, M., Mescheder, L., Niemeyer, M., Strauss, T., Geiger, A.: Texture fields: Learning texture representations in function space. In: Proceedings IEEE International Conf. on Computer Vision (ICCV) (2019)

35. Ost, J., Mannan, F., Thuerey, N., Knodt, J., Heide, F.: Neural scene graphs for dynamic scenes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2856–2865 (June 2021)

36. Park, J.J., Florence, P., Straub, J., Newcombe, R., Lovegrove, S.: Deepsdf: Learning continuous signed distance functions for shape representation. In: CVPR (2019)

37. Park, K., Patten, T., Vincze, M.: Pix2pose: Pixel-wise coordinate regression of objects for 6d pose estimation. In: ICCV (2019)

38. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S.: Pytorch: An imperative style, high-performance deep learning library. In: Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., Garnett, R. (eds.) Advances in Neural Information Processing Systems 32, pp. 8024–8035. Curran Associates, Inc. (2019)

39. Peng, S., Liu, Y., Huang, Q., Zhou, X., Bao, H.: Pvnet: Pixel-wise voting network for 6dof pose estimation. In: CVPR (2019)

40. Pitteri, G., Ramamonjisoa, M., Ilic, S., Lepetit, V.: On object symmetries and 6d pose estimation from images. In: 2019 International Conference on 3D Vision (3DV). pp. 614–622. IEEE (2019)

41. Rad, M., Lepetit, V.: Bb8: A scalable, accurate, robust to partial occlusion method for predicting the 3d poses of challenging objects without using depth. In: ICCV (2017)

42. Remelli, E., Lukoianov, A., Richter, S., Guillard, B., Bagautdinov, T., Baque, P., Fua, P.: Meshsdf: Differentiable iso-surface extraction. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M.F., Lin, H. (eds.) Advances in Neural Information Processing Systems. vol. 33, pp. 22468–22478. Curran Associates, Inc. (2020), https://proceedings.neurips.cc/paper/2020/file/fe40fb944ee700392ed51bfe84dd4e3d-Paper.pdf

43. Shugurov, I., Zakharov, S., Ilic, S.: Dpodv2: Dense correspondence-based 6 dof pose estimation. TPAMI (2021)

44. Sitzmann, V., Martel, J., Bergman, A., Lindell, D., Wetzstein, G.: Implicit neural representations with periodic activation functions. NeurIPS (2020)

45. Sitzmann, V., Zollhoefer, M., Wetzstein, G.: Scene representation networks: Continuous 3d-structure-aware neural scene representations. NeurIPS (2019)

46. Sundermeyer, M., Marton, Z.C., Durner, M., Brucker, M., Triebel, R.: Implicit 3d orientation learning for 6d object detection from rgb images. In: ECCV (2018)

47. Sundermeyer, M., Marton, Z.C., Durner, M., Triebel, R.: Augmented autoencoders: Implicit 3D orientation learning for 6d object detection. International Journal of Computer Vision **128**(3), 714–729 (2020)

48. Takikawa, T., Litalien, J., Yin, K., Kreis, K., Loop, C., Nowrouzezahrai, D., Jacobson, A., McGuire, M., Fidler, S.: Neural geometric level of detail: Real-time rendering with implicit 3d shapes. In: CVPR (2021)

49. Tejani, A., Tang, D., Kouskouridas, R., Kim, T.K.: Latent-class hough forests for 3d object detection and pose estimation. In: European Conference on Computer Vision. pp. 462–477. Springer (2014)

50. Tekin, B., Sinha, S.N., Fua, P.: Real-time seamless single shot 6d object pose prediction. In: CVPR (2018)

51. Tian, M., Ang, M.H., Lee, G.H.: Shape prior deformation for categorical 6d object pose and size estimation. In: European Conference on Computer Vision. pp. 530–546. Springer (2020)

52. Wang, C., Martín-Martín, R., Xu, D., Lv, J., Lu, C., Fei-Fei, L., Savarese, S., Zhu, Y.: 6-pack: Category-level 6d pose tracker with anchor-based keypoints. In: 2020 IEEE International Conference on Robotics and Automation (ICRA). pp. 10059–10066. IEEE (2020)
53. Wang, C., Xu, D., Zhu, Y., Martín-Martín, R., Lu, C., Fei-Fei, L., Savarese, S.: Densefusion: 6d object pose estimation by iterative dense fusion. In: CVPR (2019)
54. Wang, H., Sridhar, S., Huang, J., Valentin, J., Song, S., Guibas, L.J.: Normalized object coordinate space for category-level 6d object pose and size estimation. In: CVPR (2019)
55. Wang, N., Zhang, Y., Li, Z., Fu, Y., Liu, W., Jiang, Y.G.: Pixel2mesh: Generating 3d mesh models from single rgb images. In: Proceedings of the European conference on computer vision (ECCV). pp. 52–67 (2018)
56. Wen, B., Bekris, K.E.: Bundletrack: 6d pose tracking for novel objects without instance or category-level 3d models. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (2021)
57. Xiang, Y., Schmidt, T., Narayanan, V., Fox, D.: Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. In: RSS (2018)
58. Yuan, W., Khot, T., Held, D., Mertz, C., Hebert, M.: PCN: Point completion network. In: 3D Vision (3DV), 2018 International Conference on (2018)
59. Zakharov, S., Kehl, W., Bhargava, A., Gaidon, A.: Autolabeling 3d objects with differentiable rendering of sdf shape priors. In: CVPR (2020)
60. Zhang, C., Cui, Z., Zhang, Y., Zeng, B., Pollefeys, M., Liu, S.: Holistic 3d scene understanding from a single image with implicit representation. In: CVPR (2021)
61. Zhou, X., Wang, D., Krähenbühl, P.: Objects as points. arXiv preprint arXiv:1904.07850 (2019)
62. Zhou, Y., Barnes, C., Lu, J., Yang, J., Li, H.: On the continuity of rotation representations in neural networks. In: CVPR (2019)