SDFit: 3D Object Pose and Shape by Fitting a Morphable SDF to a Single Image
Supplementary Material

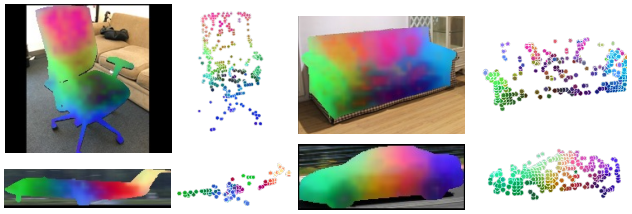

Figure S.1. Feature matching examples. For each pair – Left: PCA color-coded image features ($\mathcal{F}_\mathcal{I}$). Right: Corresponding mSDF 3D points ($\mathcal{F}_S$), colored according to matched image pixels.

S.1. 2D-3D Pixel-Vertex Matching

The task of single-image 3D pose and shape estimation presents significant challenges due to depth ambiguities, and (self-)occlusions. To address these issues, we propose a zero-shot pose initialization technique leveraging deep foundational features [49, 76], inspired by image-to-image (2D-2D) matching methods [74].

Starting from a shape initialization obtained via our procedure (see Sec. 3.3), the goal is to establish 2D-to-3D correspondences by matching 2D pixels to 3D points of the mSDF. Using a pre-trained ControlNet [76] and DINOv2 [49] model we extract feature descriptors for the 2D image, $\mathcal{F}_\mathcal{I}$, and 3D shape, $\mathcal{F}_S$, as detailed in Sec. 3.4. These descriptors are matched via cosine similarity (Eq. (9)) to obtain a set of 2D-to-3D pixel-vertex correspondences.

By leveraging the semantic and geometric cues encoded in the features of ControlNet and DINOv2 [4], our approach implicitly identifies the visible 3D vertices from 2D pixels. Examples of these matches are shown in Fig. S.1, where these are color-coded via the PCA of $\mathcal{F}_\mathcal{I}$.

S.2. Occlusion Sensitivity

As discussed in Sec. 4.3 in paragraph "Shape Reconstruction under Occlusion," we evaluate robustness under occlusion by performing a sensitivity analysis against ZeroShape [28]. Specifically, we augment Pix3D [61] test images by randomly rendering rectangle occluders covering varying percentages (from 10% to 60%) of the object bounding box; see examples in Fig. 7.

In the main paper we report the results in a plot (Fig. 8). Here we report the numerical values that correspond to this plot in terms of the Chamfer Distance metric – see Tab. S.1.

SDFit consistently outperforms ZeroShape for all occlusion levels (both in terms of mean error and st. dev.), preserving object coherence even with substantial occlusion. Notably, ZeroShape struggles even with minor occlusions (10%-20%), emphasizing SDFit's practical advantage.

Occlusion	Pix3D (mean CD@XX) ↓	
(%)	ZeroShape [28]	SDFit (Ours)
0%	**3.44**±1.45	3.53±**0.82**
10%	3.80±1.42	**3.66**±**1.03**
20%	4.69±1.20	**3.65**±**0.99**
30%	5.53±1.17	**3.82**±**1.00**
40%	6.40±1.53	**3.74**±**1.13**
50%	6.76±1.83	**3.74**±**1.23**
60%	7.45±2.48	**3.83**±**1.15**

Table S.1. Sensitivity analysis on occlusion. We evaluate reconstruction accuracy under varying occlusion levels on the Pix3D [61] test set, reporting the mean and standard deviation of Chamfer Distance (CD). We also show the case with 0% occlusion (result from Tab. 3) as reference. Note that the occlusion percentage is computed on bounding boxes (that might be non-tight for the depicted object), so 60% corresponds to excessively strong occlusions; see examples in Fig. 7. SDFit consistently outperforms ZeroShape (ZS), demonstrating greater stability and robustness as occlusion increases, whereas ZeroShape heavily deteriorates.

S.3. Ablation of SDFit Modules

We replace our shape- and pose-estimation modules with GT information, and report the 2D IoU (%) on the Pix3D dataset similar to Tab. 4.

We compare three methods: (1) `SDFit` that refines both shape and pose and achieves an IoU of 84.3%, (2) `SDFit-poseGT` that refines only shape and achieves 85.6%, and (3) `SDFit-shapeGT` that refines only pose and achieves 79.4%.

This shows that SDFit performs on par with privileged baselines. All variants clearly outperform `ZeroShape+RnC` that achieves 73.3%.

S.4. Discussion & Future Work

We leverage foundational features for pose initialization. As common in existing work [75], sometimes there might be potential left-right ambiguities that we tackle by evaluating two vertically mirrored candidates. Future work will explore more involved approaches, e.g., via learned regression or by directly lifting 2D features into 3D via metric depth [5].

Moreover, sometimes fine details may be missed, as in other neural-field-based methods [12, 28], due to the fixed resolution grid used for mesh extraction. Future work will look into dynamically adapting resolution, or enhancing the mSDF expressiveness with a more "flexible" latent space.