Neural Correspondence Field for Object Pose Estimation

Lin Huang[1*] Tomas Hodan[2] Lingni Ma[2] Linguang Zhang[2]
Luan Tran[2] Christopher Twigg[2] Po-Chen Wu[2]
Junsong Yuan[1] Cem Keskin[2] Robert Wang[2]

[1]University at Buffalo [2]Reality Labs at Meta

Abstract. We propose a method for estimating the 6DoF pose of a rigid object with an available 3D model from a single RGB image. Unlike classical correspondence-based methods which predict 3D object coordinates at pixels of the input image, the proposed method predicts 3D object coordinates at 3D query points sampled in the camera frustum. The move from pixels to 3D points, which is inspired by recent PIFu-style methods for 3D reconstruction, enables reasoning about the whole object, including its (self-)occluded parts. For a 3D query point associated with a pixel-aligned image feature, we train a fully-connected neural network to predict: (i) the corresponding 3D object coordinates, and (ii) the signed distance to the object surface, with the first defined only for query points in the surface vicinity. We call the mapping realized by this network as *Neural Correspondence Field*. The object pose is then robustly estimated from the predicted 3D-3D correspondences by the Kabsch-RANSAC algorithm. The proposed method achieves state-of-the-art results on three BOP datasets and is shown superior especially in challenging cases with occlusion. The project website is at: linhuang17.github.io/NCF.

1 Introduction

Estimating the 6DoF pose of a rigid object is a fundamental computer vision problem with great importance to application fields such as augmented reality and robotic manipulation. In recent years, the problem has received considerable attention and the state of the art has improved substantially, yet there remain challenges to address, particularly around robustness to object occlusion [24, 25].

Recent PIFu-style methods for 3D reconstruction from an RGB image [66, 67, 28, 84, 37] rely on 3D implicit representations and demonstrate the ability to learn and incorporate strong priors about the invisible scene parts. For example, PIFu [66] is able to faithfully reconstruct a 3D model of the whole human body, and DRDF [37] is able to reconstruct a 3D model of the whole indoor scene, including parts hidden behind a couch. Inspired by these results, we propose a 6DoF object pose estimation method based on a 3D implicit representation and analyze its performance specifically in challenging cases with occlusion.

[*]Work done during Lin Huang's internship with Reality Labs at Meta.

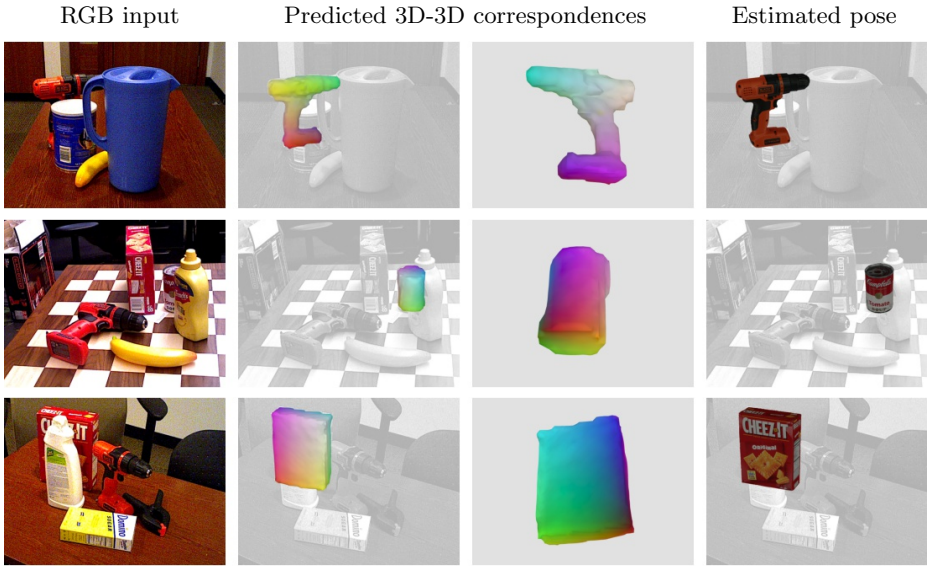


Fig. 1. **An overview of the proposed method.** The object pose is estimated from 3D-3D correspondences established by predicting 3D object coordinates at 3D query points densely sampled in the camera frustum. For efficient selection of reliable correspondences nearby the object surface, the method predicts for each query point also the signed distance to the surface. The middle columns show two views of a mesh that is reconstructed by Marching Cubes [45] from the predicted signed distances and colored with the predicted 3D object coordinates (the mesh is reconstructed only for visualization purposes, not when estimating the object pose). The 3D CAD model, which is assumed available for each object, is shown in the estimated pose on the right.

Similarly to PIFu [66], the proposed method makes predictions for 3D query points that are sampled in the camera frustum and associated with pixel-aligned image features. PIFu predicts color and occupancy, *i.e.*, a binary signal that indicates whether a query point is inside or outside the object. Instead, the proposed method predicts (i) the corresponding 3D object coordinates, and (ii) the signed distance to the object surface, with the first defined only for query points in the surface vicinity, *i.e.*, points for which the predicted signed distance is below a threshold. The 6DoF object pose is then robustly estimated from the predicted 3D-3D correspondences between 3D query points and the predicted 3D object coordinates by the Kabsch algorithm [31] in combination with RANSAC [16].

Classical methods for 6DoF object pose estimation [9, 5, 59, 83, 63, 73, 61, 23] rely on 2D-3D correspondences established between pixels of the input image and the 3D object model, and estimate the pose by the PnP-RANSAC algorithm [39]. The proposed method predicts 3D object coordinates for 3D query points instead of pixels. This enables reasoning about the whole object surface, including self-occluded parts and parts occluded by other objects. In Sec. 5, we show that the proposed method noticeably outperforms a baseline method

that relies on the classical 2D-3D correspondences. Besides, we show that the proposed method outperforms all existing methods with the same training and evaluation setup (*i.e.*, RGB-only and without any iterative refinement of pose estimates) on datasets YCB-V, LM-O, and LM from the BOP benchmark [24, 25].

This work makes the following contributions:

1. The first method for 6DoF object pose estimation which demonstrates the effectiveness of a 3D implicit representation in solving this problem.
2. Neural Correspondence Field (NCF), a learned 3D implicit representation defined by a mapping from the camera space to the object model space, is used to establish 3D-3D correspondences from a single RGB image.
3. The proposed method noticeably outperforms a baseline based on 2D-3D correspondences and achieves state-of-the-art results on three BOP datasets.

2 Related Work

6DoF Object Pose Estimation. Early methods for 6DoF object pose estimation assumed a grayscale or RGB input image and relied on local image features [46, 9] or template matching [7]. After the introduction of Kinect-like sensors, methods based on RGB-D template matching [21, 26], point-pair features [15, 22, 78], 3D local features [19], and learning-based methods [5, 72, 35] demonstrated superior performance over RGB-only counterparts. Recent methods are based on convolutional neural networks (CNNs) and focus primarily on estimating the pose from RGB images. In the 2020 edition of the BOP challenge [25], CNN-based methods finally caught up with methods based on point-pair features which were dominating previous editions of the challenge. A popular approach adopted by the CNN-based methods is to establish 2D-3D correspondences by predicting 3D object coordinates at densely sampled pixels, and robustly estimate the object pose by the PnP-RANSAC algorithm [30, 79, 83, 59, 41, 23]. In Sec. 5, we show that our proposed method outperforms a baseline method that follows the 2D-3D correspondence approach and shares implementation of the common parts with the proposed method. Methods establishing the correspondences in the opposite direction, *i.e.*, by predicting the 2D projections of a fixed set of 3D keypoints pre-selected for each object model, have also been proposed [63, 60, 54, 73, 76, 27, 61]. Other approaches localize the objects with 2D bounding boxes, and for each box predict the pose by regression [80, 40, 47, 38] or classification into discrete viewpoints [33, 10, 71]. However, in the case of occlusion, estimating accurate 2D bounding boxes covering the whole object, including the invisible parts, is problematic [33].

Shape Reconstruction with Implicit Representations. Recent works have shown that a 3D shape can be modeled by a continuous and differentiable implicit representation realized by a fully-connected neural network. Examples of such representations include signed distance fields (SDF) [57, 17, 1, 2, 69], which map a 3D query point to the signed distance from the surface, and binary occupancy fields [48, 8, 42], which map a 3D query point to the occupancy value. Following

the success of implicit representations, GraspingField [32] extends the idea to reconstructing hands grasping objects. Instead of learning a single SDF, the method learns one SDF for hand and one for object, which allows to directly enforce physical constraints such as no interpenetration and proper contact.

For image-based reconstruction, Texture fields [55] learn textured 3D models by mapping a shape feature, an image feature, and a 3D point to color. OccNet [48] proposes to condition occupancy prediction on an image feature extracted by a CNN. DISN [81] improves this technique by combining local patch features with a global image feature to estimate SDF for 3D query points. PIFu [66], which is closely related to our work, first extracts an image feature map by an hourglass CNN and then applies a fully-connected neural network to map a pixel-aligned feature with the depth of a 3D query point to occupancy. The follow-up work, PIFuHD [67], recovers more detailed geometry by leveraging the surface normal map and multi-resolution volumes. PIFu and PIFuHD focus on human digitization. As for many other methods, experiments are done on images with cleanly segmented foreground. Recently, NeRF-like methods reported impressive results in scene modeling [70, 44, 53, 82, 50]. These methods typically require multi-view images with known camera calibration. For an in-depth discussion, we refer to the survey in [74]. In this work, we focus on a single input image and reconstruct known objects in unknown poses that we aim to recover.

Learning Dense Correspondences. One of the pioneering works that learns dense correspondences is proposed in [68] for camera relocalization, and extended for pose estimation of specific rigid objects in [5, 6, 49]. These methods predict 3D scene/object coordinates at each pixel of the input image by a random forest. Later methods predict the coordinates by a CNN [30, 59, 41, 83, 23]. NOCS [79] defines normalized object coordinates for category-level object pose estimation. Besides correspondences for object pose estimation, DensePose [18] densely regresses part-specific UV coordinates for human pose estimation. CSE [51] extends the idea to predict correspondences for deformable object categories by regressing Laplace-Beltrami basis and is extended to model articulated shapes in [36]. These methods focus on learning mapping from pixels to 3D coordinates. DIF-Net [12] jointly learns the shape embedding of an object category and 3D-3D correspondences with respect to a template. Similarly, NPMs [56] learns a 3D deformation field to model deformable shapes. Recent methods [62, 58, 77] model deformable shapes by learning radiance and deformation fields. None of these methods aims to recover the pose from images.

3 Preliminaries

Notations. An RGB image is denoted by $I : \mathbb{R}^2 \mapsto \mathbb{R}^3$ and can be mapped to a feature map $F : \mathbb{R}^2 \mapsto \mathbb{R}^K$ with K channels by an hourglass neural network [52, 66]. A 3D point $\mathbf{x} = [x, y, z]^\top \in \mathbb{R}^3$ in the camera coordinate frame can be projected to a pixel $[u, v]^\top \in \mathbb{R}^2$ by the projection function $\pi(\mathbf{x}) : \mathbb{R}^3 \mapsto \mathbb{R}^2$. Without loss of generality, we use a pinhole camera model with the projection

function defined as: $\pi(\mathbf{x}) = [xf_x/z + c_x,\ yf_y/z + c_y]^\top$, where f_x, f_y is the focal length and (c_x, c_y) is the principal point.

A 6DoF object pose is defined as a rigid transformation $(R, \mathbf{t})$, where $R \in \mathbb{SO}(3)$ is a 3D rotation matrix and $\mathbf{t} \in \mathbb{R}^3$ is a 3D translation vector. A 3D point $\mathbf{y}$ in the model coordinate frame (also referred to as *3D object coordinates* [5]) is transformed to a 3D point $\mathbf{x}$ in the camera coordinate frame as: $\mathbf{x} = R\mathbf{y} + \mathbf{t}$.

Signed Distance Function (SDF) [11, 57]. In the proposed method, the object surface is represented implicitly with a signed distance function, $\psi(\mathbf{x}) : \mathbb{R}^3 \mapsto \mathbb{R}$, which maps a 3D point $\mathbf{x}$ to the signed distance between $\mathbf{x}$ and the object surface. The signed distance is zero on the object surface, positive if $\mathbf{x}$ is outside the object and negative if $\mathbf{x}$ is inside.

Kabsch Algorithm [31]. Given $N \geq 3$ pairs of corresponding 3D points $X = \{\mathbf{x}_i\}_N$ and $Y = \{\mathbf{y}_i\}_N$, the Kabsch algorithm finds a rigid transformation that aligns the corresponding 3D points by minimizing the following least squares:

$$R^\star, \mathbf{t}^\star = \arg\min_{R,\mathbf{t}} \sum_i^N \|R\mathbf{y}_i + \mathbf{t} - \mathbf{x}_i\|_2 \,. \tag{1}$$

The 3D rotation is solved via SVD of the covariance matrix: $USV^\top = \text{Cov}(X - \mathbf{c}_X, Y - \mathbf{c}_Y)$, $R^\star = VU^\top$, where $\mathbf{c}_X$ and $\mathbf{c}_Y$ are centroids of the point sets X and Y respectively. To ensure right-handed coordinate system, the signs of the last column of matrix V are flipped if $\det(R^\star) = -1$ [31]. The 3D translation is then calculated as: $\mathbf{t}^\star = \mathbf{c}_X - R^\star\mathbf{c}_Y$. In the proposed method, we combine the Kabsch algorithm with a RANSAC-style fitting scheme [16] to estimate the object pose from 3D-3D correspondences.

PIFu [66]. The PIFu method reconstructs 3D models of humans from segmented single/multi-view RGB images. For the single-view inference, the method first obtains a feature map F with an hourglass neural network. Then it applies a fully-connected neural network, $f_{\text{PIFu}}(F(\pi(\mathbf{x})), \mathbf{x}_z) = o$, to map a pixel-aligned feature $F(\pi(\mathbf{x}))$ and the depth $\mathbf{x}_z$ of a 3D query point $\mathbf{x}$ to the occupancy $o \in [0, 1]$ (1 means the 3D point is inside the model and 0 means it is outside).

4 The Proposed Method

This section describes the proposed method for estimating the 6DoF object pose from an RGB image. The image is assumed to show a single target object, potentially with clutter, occlusion, and diverse lighting and background. In addition, the 3D object model, camera intrinsic parameters, and a large set of training images annotated with ground-truth object poses are assumed available.

The proposed method consists of two stages: (1) prediction of 3D-3D correspondences between the camera coordinate frame and the model coordinate frame (Sec. 4.1), and (2) fitting the 6DoF object pose to the predicted correspondences using the Kabsch-RANSAC algorithm (Sec. 4.2).

4.1 Predicting Dense 3D-3D Correspondences

Neural Correspondence Field (NCF). The 3D-3D correspondences are established using NCF defined as a mapping from the pixel-aligned feature $F(\pi(\mathbf{x}))$ and the depth $\mathbf{x}_z$ of a 3D query point $\mathbf{x}$ in the camera frame to the corresponding 3D point $\mathbf{y}$ in the model frame and its signed distance s (see also Fig. 2):

$$f_{\text{NCF}} : \mathbb{R}^K \times \mathbb{R} \mapsto \mathbb{R}^3 \times \mathbb{R} \quad \text{as} \quad f_{\text{NCF}}\big(F(\pi(\mathbf{x})), \mathbf{x}_z; \boldsymbol{\theta}\big) = \big(\mathbf{y}, s\big), \tag{2}$$

where $\boldsymbol{\theta}$ are parameters of a fully-connected neural network f_{NCF} that realizes the mapping. In our experiments, f_{NCF} has the same architecture as the fully-connected network in PIFu [66], except the output dimension is 4 and tanh is used as an activation function in the last layer, as in [57]. The feature extractor F is realized by the hourglass neural network from PIFu and is applied to the input image remapped to a reference pinhole camera (arbitrarily chosen). The remapping is important to keep the depth $\mathbf{x}_z$ in accord with the image feature $F(\pi(\mathbf{x}))$ across images captured by cameras with different focal lengths.

Compared to PIFu, NCF additionally predicts the corresponding 3D point $\mathbf{y}$, which enables establishing 3D-3D correspondences that are used for object pose fitting. Besides, NCF predicts the signed distance instead of the binary occupancy. This enables efficient selection of near-surface correspondences by thresholding the signed distances. Using near-surface correspondences increases pose fitting accuracy as learning correspondences from images with diverse background becomes ill-posed for 3D points far from the surface. Since the 3D object model is available, the signed distance s could be calculated from the predicted 3D point $\mathbf{y}$. However, we chose to predict the signed distance explicitly to speed up the method at both training and test time (predicting the value explicitly takes virtually no extra time nor resources).

Training. With parameters of the hourglass network F denoted as $\boldsymbol{\eta}$ and parameters of the NCF network f_{NCF} denoted as $\boldsymbol{\theta}$, the two networks are trained

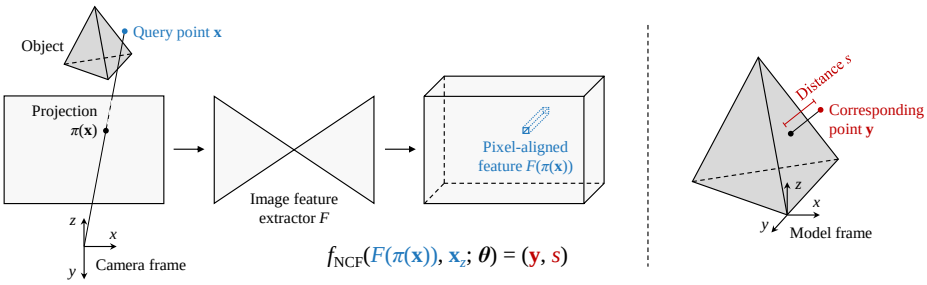


Fig. 2. **Neural Correspondence Field** is a mapping learned by a fully-connected neural network f_{NCF} with parameters $\boldsymbol{\theta}$. The input of the network is (i) an image feature $F(\pi(\mathbf{x}))$ extracted at the 2D projection $\pi(\mathbf{x})$ of a 3D query point $\mathbf{x}$ sampled in the camera frustum, and (ii) the depth $\mathbf{x}_z$ of $\mathbf{x}$. The output is (i) the corresponding 3D point $\mathbf{y}$ in the model frame, and (ii) the signed distance s between $\mathbf{y}$ and the object surface. The point $\mathbf{y}$ is defined only if $|s|$ is below a fixed clamping threshold δ.

jointly by solving the following optimization problem:

$$\boldsymbol{\eta}^{\star}, \boldsymbol{\theta}^{\star} = \arg\min_{\boldsymbol{\eta},\boldsymbol{\theta}} L_{\mathbf{y}} + \lambda L_s, \tag{3}$$

where $L_{\mathbf{y}}$ and L_s are regression losses on the 3D point $\mathbf{y}$ and the signed distance s, respectively. The scalar λ is a balancing weight. Assuming N 3D points sampled in the camera frustum per image, the losses are defined as:

$$L_{\mathbf{y}} = \min_{(\bar{R},\bar{\mathbf{t}})\in S} \frac{1}{N} \sum_i \mathbb{1}\left(|\psi\left(\bar{\mathbf{y}}_i\right)| < \delta\right) H\left(\bar{R}\mathbf{y}_i + \bar{\mathbf{t}}, \mathbf{x}_i\right), \tag{4}$$

$$L_s = \frac{1}{N} \sum_i \left|\text{clamp}\left(\psi(\bar{\mathbf{y}}_i), \delta\right) - \text{clamp}\left(s, \delta\right)\right|, \tag{5}$$

where $(\bar{R}, \bar{\mathbf{t}})$ is a ground-truth pose, $\bar{\mathbf{y}}_i = \bar{R}^{-1}(\mathbf{x}_i - \bar{\mathbf{t}})$ is the ground-truth corresponding 3D point, and δ is a clamping parameter controlling the distance from the surface over which we expect to maintain a metric SDF, as in [57]. The indicator function $\mathbb{1}(\cdot)$ selects points within the clamping distance, and H is the Huber loss [29]. To handle symmetric objects, we adopt the approach from NOCS [79] which uses a pre-defined set of symmetry transformations (continuous symmetries are discretized) to get a set S of possible ground-truth poses.

Sampling 3D Query Points. Given a training image and the ground-truth object pose, the 3D object model is first transformed to the camera frame to assist with sampling of the query points. As the training images may show the object in diverse scenes, we found it crucial to focus the training on the object by sampling the query points more densely around the object surface. In our experiments, we first sample three types of points: 12500 points nearby the surface, 1000 points inside the bounding sphere of the model, and 1000 points inside the camera frustum. From these points, we sample 2500 points inside the model and 2500 points outside. Note that this sampling strategy is invariant to occlusion, which forces the network f_{NCF} to learn the complete object surface. At test time, with no knowledge of the object pose, the points are sampled at centers of voxels that fill up the camera frustum in a specified depth range.

4.2 Pose Fitting

To estimate the 6DoF object pose at test time, a set of 3D-3D correspondences, $C = \{(\mathbf{x}_i, \mathbf{y}_i)\}_M$ with $M \geq 3$, is established by linking each 3D query point $\mathbf{x}$ with the predicted 3D point $\mathbf{y}$ for which the predicted signed distance s is below the threshold δ. The object pose is then estimated from C by a RANSAC-style fitting scheme [16], which iteratively proposes a pose hypothesis by sampling a random triplet of 3D-3D correspondences from C and calculating the pose from the triplet by the Kabsch algorithm detailed in Sec. 3. The quality of a pose hypothesis $(R, \mathbf{t})$ is measured by the number of inliers, *i.e.*, the number of correspondences $(\mathbf{x}, \mathbf{y}) \in C$ for which $\|R\mathbf{y} + \mathbf{t} - \mathbf{x}\|_2$ is below a fixed threshold τ. In the presented experiments, a fixed number of pose hypotheses is generated

for each test image, and the final pose estimate is given by the hypothesis of the highest quality which is further refined by the Kabsch algorithm applied to all inliers. Note that the pose is not estimated at training time as the pose estimate is not involved in the training loss calculation.

Since we assume that a single instance of the object of interest is present in the input image, the set C is assumed to contain only correspondences originating from the single object instance, while being potentially contaminated with outlier correspondences caused by errors in prediction. The method could be extended to handle multiple instances of the same object, *e.g.*, by using the Progressive-X multi-instance fitting scheme [4], as in EPOS [23].

5 Experiments

This section analyzes the proposed method for 6DoF object pose estimation and compares its performance with the state-of-the-art methods from the BOP Challenge 2020 [25]. To demonstrate the advantage of predicting dense 3D-3D correspondences, the proposed method is also compared with a baseline that relies on classical 2D-3D correspondences.

5.1 2D-3D Baseline Method

Many state-of-the-art methods for 6DoF object pose estimation build on 2D-3D correspondence estimation [5, 79, 83, 59, 41, 23]. While these methods are included in overall evaluation, we also design a directly comparable baseline that uses the same architecture of the feature extractor F and of the subsequent fully-connected network as the proposed method described in Sec. 4. However, unlike the fully-connected network f_{NCF} which takes a pixel-aligned feature $F(\pi(\mathbf{x}))$ and the depth $\mathbf{x}_z$ of a 3D query point $\mathbf{x}$ and outputs the corresponding 3D coordinates $\mathbf{y}$ and the signed distance s, the baseline method relies on a network f_{BL} which takes only a pixel-aligned feature $F(\mathbf{p})$ at a pixel $\mathbf{p}$ and outputs the corresponding 3D coordinates $\mathbf{y}$ and the probability $q \in [0, 1]$ that the object is present at $\mathbf{p}$: $f_{\text{BL}} : \mathbb{R}^K \mapsto \mathbb{R}^3 \times \mathbb{R}$ as $f_{\text{BL}}(F(\mathbf{p}); \boldsymbol{\theta}) = (\mathbf{y}, q)$. The baseline method is trained by solving the following optimization problem:

$$\boldsymbol{\eta}^\star, \boldsymbol{\theta}^\star = \arg\min_{\boldsymbol{\eta},\boldsymbol{\theta}} L_{\mathbf{y}} + \lambda L_q \tag{6}$$

$$= \arg\min_{\boldsymbol{\eta},\boldsymbol{\theta}} \min_{(\bar{R},\bar{\mathbf{t}})\in S} \frac{1}{U} \sum_i \bar{q}_i H\left(\bar{R}\mathbf{y}_i + \bar{\mathbf{t}}, \mathbf{x}_i\right) + \lambda \frac{1}{U} \sum_i E\left(q_i, \bar{q}_i\right), \tag{7}$$

where U is the number of pixels, E is the softmax cross entropy loss, $\bar{q}$ is given by the ground-truth object mask, and $\bar{\mathbf{y}}$ are the ground-truth 3D coordinates. At test time, 2D-3D correspondences are established at pixels with $q > 0.5$ and used to fit the object pose with the PnP-RANSAC algorithm [39]. In RANSAC, a 2D-3D correspondence $(\mathbf{p}, \mathbf{y})$ is considered an inlier *w.r.t.* a pose hypothesis $(R, \mathbf{t})$ if $\|\mathbf{p} - \pi(R\mathbf{y} + \mathbf{t})\|_2$ is below a fixed threshold τ_{2D}.

We experiment with two variants of the baseline: "Baseline-visib" defines $\bar{q} = 1$ for the visible foreground pixels, and "Baseline-full" defines $\bar{q} = 1$ for all pixels in the object silhouette, even if occluded by other objects.

5.2 Experimental Setup

Evaluation Protocol. We follow the evaluation protocol of the BOP Challenge 2020 [25]. In short, a method is evaluated on the 6DoF object localization problem, and the error of an estimated pose *w.r.t.* the ground-truth pose is calculated by three pose-error functions: Visible Surface Discrepancy (VSD) which treats indistinguishable poses as equivalent by considering only the visible object part, Maximum Symmetry-Aware Surface Distance (MSSD) which considers a set of pre-identified global object symmetries and measures the surface deviation in 3D, and Maximum Symmetry-Aware Projection Distance (MSPD) which considers the object symmetries and measures the perceivable deviation. An estimated pose is considered correct *w.r.t.* a pose-error function e, if $e < \theta_e$, where $e \in \{\text{VSD}, \text{MSSD}, \text{MSPD}\}$ and θ_e is the threshold of correctness. The fraction of annotated object instances for which a correct pose is estimated is referred to as Recall. The Average Recall *w.r.t.* a function e, denoted as AR_e, is defined as the average of the Recall rates calculated for multiple settings of the threshold θ_e and also for multiple settings of a misalignment tolerance τ in the case of VSD. The overall accuracy of a method is measured by the Average Recall: $\text{AR} = (\text{AR}_{\text{VSD}} + \text{AR}_{\text{MSSD}} + \text{AR}_{\text{MSPD}}) / 3$.

The BOP Challenge 2020 considers the problem of 6DoF localization of a varying number of instances of a varying number of objects from a single image. To evaluate the proposed method, which was designed to handle a single instance of a single object, we consider only BOP datasets where images show up to one instance of each object. On images that show single instances of multiple objects, we evaluate the proposed method multiple times, each time estimating the pose of a single object instance using the neural networks trained for that object.

Datasets. The experiments are conducted on the BOP 2020 [25] version of three datasets: LM [21], LM-O [5], and YCB-V [80]. The datasets include color 3D object models and RGB-D images of VGA resolution annotated with ground-truth 6DoF object poses (only the RGB channels are used in this work). LM contains 15 texture-less objects with discriminative color, shape, and size. Every object is associated with a set of 200 test images, each showing one annotated object instance under significant clutter and no or mild occlusion. LM-O provides ground-truth annotation for instances of eight LM objects in one of the test sets, which introduces challenging test cases with various levels of occlusion. YCB-V includes 21 objects that are both textured and texture-less, 900 test images showing the objects with occasional occlusions and limited clutter, and 113K real and 80K OpenGL-rendered training images. Each of these datasets is also associated with 50K physically-based rendered (PBR) images generated by BlenderProc [14, 13] and provided by the BOP organizers. The datasets provide also sets of object symmetry transformations that are used in Eq. 5 and 7.

Training. We report results achieved by the proposed and the baseline methods trained on the synthetic PBR images. On the YCB-V dataset, for which real training images are available, we report also results achieved by the proposed method trained on both real and synthetic PBR images. To reduce the domain gap between the synthetic training and real test images, the training images are augmented by randomly adjusting contrast, brightness, sharpness, and color, as in [38]. The feature extractor F and networks f_{NCF} and f_{BL} are initialized with random weights. The networks are optimized by RMSProp [75] with the batch size of 4 training images, learning rate of 0.0001, no learning rate drop, and the balancing weight λ set to 1. On LM and LM-O, the optimization is run for 220 epochs. On YCB-V, the optimization is run for 300 epochs on synthetic PBR images, and then for extra 150 epochs on PBR and real images (we report scores before and after the extra epochs). Special neural networks are trained for each object, while all hyper-parameters are fixed across all objects and datasets.

Method Parameters. The architecture of neural networks is adopted from PIFu [66]. Specifically, the feature extractor F is a stacked hourglass network with the output stride of 4 and output channel of 256. Networks f_{NCF} and f_{BL} have four hidden fully-connected layers with 1024, 512, 256 and 128 neurons and with skip connections from F. Unless stated otherwise, the clamping distance $\delta = 5\,\text{mm}$, the inlier threshold $\tau_{\text{3D}} = 20\,\text{mm}$, the inlier threshold for the baseline method $\tau_{\text{2D}} = 4\,\text{px}$, and the RANSAC-based pose fitting in the proposed and the baseline method is run for a fixed number of 200 iterations. The sampling step of 3D query points at test time is 10 mm (in all three axes) and the near and far planes of the camera frustum, in which the points are sampled, is determined by the range of object distances annotated in the test images (the BOP benchmark explicitly allows using this information at test time). We converged to these settings by experimenting with different parameter values and optimizing the performance of both the proposed and the baseline method.

In the presented experiments, the signed distance $\psi(\mathbf{y})$ is measured from the query point $\mathbf{x}$ to the closest point on the model surface along the projection ray (*i.e.*, a ray passing through the camera center and $\mathbf{x}$), not to the closest point in 3D as in the conventional SDF [57]. However, our additional experiments suggest the two definitions yield comparable performance.

5.3 Main Results

Accuracy. Tab. 1 compares the proposed method (NCF) with participants of the BOP Challenge 2020 and with the baseline method described in Sec. 5.1. On the YCB-V dataset, NCF trained on the synthetic PBR images outperforms all competitors which also rely only on RGB images and which do not apply any iterative refinement to the pose estimates. NCF achieves 17.4% absolute improvement over EPOS [23] and 28.3% over CDPNv2 [41], which are trained on the same set of PBR images, and 13.0% and up over [43, 41, 59], which are trained on PBR and real images. Training on the additional real images improves the AR score of NCF further to 77.5. Although with smaller margins, NCF outperforms

Method	Train	..type	Test	Refine	YCB-V	..time	LM-O	..time
NCF (ours)	rgb	pbr	rgb	–	**67.3**	1.09	**63.2**	4.33
Baseline-full	rgb	pbr	rgb	–	37.1	0.74	33.9	0.81
Baseline-visib	rgb	pbr	rgb	–	31.9	0.71	31.6	0.79
EPOS [23]	rgb	pbr	rgb	–	49.9	0.76	54.7	0.47
CDPNv2 [41]	rgb	pbr	rgb	–	39.0	0.45	62.4	0.16
NCF (ours)	rgb	pbr+real	rgb	–	**77.5**	1.09	**63.2**	4.33
leaping 2D-6D [43]	rgb	pbr+real	rgb	–	54.3	0.13	52.5	0.94
CDPNv2 [41]	rgb	pbr+real	rgb	–	53.2	0.14	62.4	0.16
Pix2Pose [59]	rgb	pbr+real	rgb	–	45.7	1.03	36.3	1.31
CosyPose [38]	rgb	pbr+real	rgbd	rc+icp	86.1	2.74	71.4	8.29
CosyPose [38]	rgb	pbr+real	rgb	rc	82.1	0.24	63.3	0.55
Pix2Pose [59]	rgb	pbr+real	rgbd	icp	78.0	2.59	58.8	5.19
FFB6D [20]	rgbd	pbr	rgbd	–	75.8	0.20	68.7	0.19
König-Hybrid [34]	rgb	syn+real	rgbd	icp	70.1	2.48	63.1	0.45
CDPNv2 [41]	rgb	pbr+real	rgbd	icp	61.9	0.64	63.0	0.51
CosyPose [38]	rgb	pbr	rgb	rc	57.4	0.34	63.3	0.55
CDPNv2 [41]	rgb	pbr	rgbd	icp	53.2	1.03	63.0	0.51
Félix&Neves [65, 64]	rgbd	syn+real	rgbd	icp	51.0	54.51	39.4	61.99
AAE [71]	rgb	syn+real	rgbd	icp	50.5	1.58	23.7	1.20
Vidal et al. [78]	–	–	d	icp	45.0	3.72	58.2	4.00
CDPN [41]	rgb	syn+real	rgb	–	42.2	0.30	37.4	0.33
Drost-3D-Only [15]	–	–	d	icp	34.4	6.27	52.7	15.95

Table 1. **Average Recall (AR) scores** on datasets YCB-V and LM-O from BOP 2020 [25]. The 2nd to 5th columns show the training and test setup: image channels used at training (*Train*), type of training images (*Train type*: *pbr* for physically-based rendered images, *syn* for synthetic images which include not only *pbr* images, *real* for real images), image channels used at test (*Test*), and type of iterative pose refinement used at test time (*Refine*: *icp* for a depth-based Iterative Closest Point algorithm, *rc* for a color-based render-and-compare refinement). While *pbr* training images are included in both datasets, *real* training images are only in YCB-V – training setups *pbr* and *pbr+real* are therefore equivalent on LM-O, which leads to several duplicate scores in the table. Top scores among methods with the same training and test setup are **bold**. The time is the average time to estimate poses of all objects in an image [s].

these competitors also on the LM-O dataset. All higher scores reported on the two datasets are achieved by methods that use the depth image or iteratively refine the estimates by ICP or a render-and-compare technique (*c.f.*, [25] for details). On the LM dataset [21] (not in Tab. 1, see BOP leaderboard [25]), NCF achieves 81.0 AR and is close the overall leading method which achives 81.4 AR and is based on point-pair features [15] extracted from depth images.

Tab. 1 also shows scores of the two variants of the baseline method. NCF achieves significant improvements over both variants, reaching almost double AR scores. As shown in Tab. 2, NCF outperforms the baseline on all objects from the three datasets. Some of the most noticeable differences are on YCB-V objects 19, 20, and 21. The baseline method struggles due to symmetries of these objects, even though it adopts a very similar symmetry-aware loss as NCF, which performs well on these objects. Qualitative results are in Fig. 4.

Speed. NCF takes 1.09 and 4.33 s on average to estimate poses of all objects in a test image from YCB-V and LM-O respectively (with a single Nvidia V100 GPU; 3–6 objects are in YCB-V images and 7–8 in LM-O images). As discussed in

Method	LM-O								LM												
	1	5	6	8	9	10'	11'	12	1	2	3'	4	5	6	7	8	9	10'	11'	12	13	14
NCF	**58**	**83**	**55**	**83**	**75**	**11**	**66**	**70**	**74**	**92**	**72**	**89**	**91**	**83**	**63**	**92**	**73**	**73**	**73**	**74**	**90**	**85**
BL-full	23	53	31	62	38	0	12	40	35	72	45	66	47	47	36	61	45	7	19	49	68	64
BL-visib	25	48	18	46	37	1	27	45	34	69	55	57	48	45	40	54	35	10	29	51	61	47

Method	YCB-V																				
	15	1'	2	3	4	5	6	7	8	9	10	11	12	13'	14	15	16'	17	18'	19'	20'	21'
NCF	**83**	**67**	**81**	**83**	**57**	**77**	**72**	**76**	**75**	**51**	**85**	**84**	**72**	**8**	**61**	**84**	**36**	**63**	**41**	**60**	**62**	**49**
BL-full	57	54	53	66	40	55	6	26	15	31	20	75	36	1	53	60	2	31	18	2	1	0
BL-visib	64	57	40	45	24	52	10	23	32	17	24	55	37	2	40	60	0	25	32	3	0	0

Table 2. **Per-object AR scores** on datasets LM-O [5], LM [21], and YCB-V [80]. Objects with symmetries are marked by the prime symbol (').

Sec. 5.4, the processing time can be decreased with sparser query point sampling or with less RANSAC iterations, both yielding only a moderate drop in AR score. Besides, NCF can be readily used for object tracking, where the exhaustive scanning of the frustum could be replaced by sampling a limited number of query points around the model in the pose estimated in the previous frame. This would require a lower number of query points and therefore faster processing.

5.4 Ablation Studies

Performance Under Occlusion. First, we study the impact of different occlusion levels on the quality of predicted 3D-3D correspondences, using the visibility information from [25]. This analysis is conducted on datasets YCB-V and LM-O which include partially occluded examples. The quality of correspondences is measured by the fraction of inliers, which is the key metric determining the success of RANSAC [16]. A correspondence $(\mathbf{x}, \mathbf{y})$ is considered an inlier if $\|\bar{R}\mathbf{y} + \bar{\mathbf{t}} - \mathbf{x}\|_2 < \tau_{\text{3D}} = 20\,\text{mm}$, where $\mathbf{x}$ is a 3D query point in the camera coordinates, $\mathbf{y}$ is the predicted 3D point in the model coordinates, and $(\bar{R}, \bar{\mathbf{t}})$ is the ground-truth object pose. Fig. 3 (left) shows the average inlier fraction for test examples split into five bins based on the object visibility. Already with around 40% visibility (*i.e.*, 60% occlusion), the established correspondences (red curve) include 20% inliers, which is typically sufficient for fitting a good pose with 200 RANSAC iterations.[1] To separately analyze the quality of correspondences established around the visible and invisible surface, we first select a subset of correspondences established at query points that are in the vicinity of the object surface in the ground-truth pose. This subset is then split into correspondences at the visible surface (green curve in Fig. 3, left) and at the invisible surface (blue curve). Although the inlier percentage is higher for correspondences at the visible surface, correspondences at the invisible surface keep up, demonstrating the ability of the proposed method to reason about the whole object.

[1] The number of required RANSAC iterations is given by $\log(1-p)/\log(1-w^n)$, where p is the desired probability of success, w is the fraction of inliers, and n is the minimal set size [16]. In the discussed case, $p = 0.8$ yields $\log(1-0.8)/\log(1-0.2^3) \approx 200$.

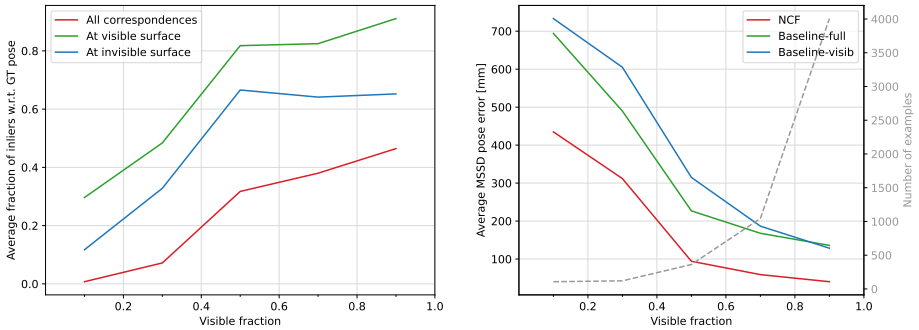
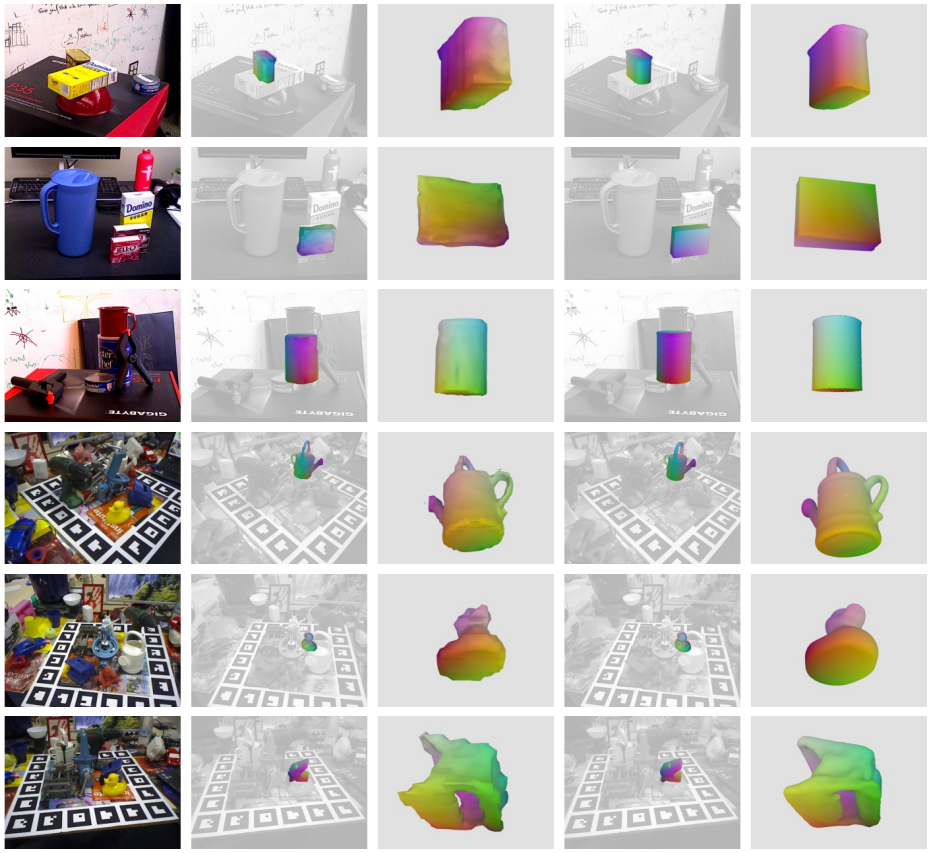


Fig. 3. **Performance w.r.t. visible object fraction.** Left: The average fraction of established 3D-3D correspondences that are inliers *w.r.t.* the ground-truth pose (*i.e.*, the error of predicted 3D object coordinates is less than a threshold $\tau_{3D} = 20\,\text{mm}$). The set of all correspondences is not the union of correspondences at the visible and invisible surface, hence the red curve is not in between the other two – see text for details. Right: The average MSSD error [25] of object poses estimated by the proposed method (NCF) and the baselines. The average values in both plots are calculated over test examples split into five bins based on the visible fraction of the object silhouette.

Next, Fig. 3 (right) shows the impact of occlusion on the average MSSD error [25] of object poses estimated by the proposed method and the baselines. The proposed method (NCF) clearly outperforms the baselines at all occlusion levels and keeps the average error below 10 cm up to around 50% occlusion.

Density of 3D Query Points. The scores discussed so far were obtained with 3D query points sampled with the step of 10 mm, *i.e.*, the points are at the centers of $10 \times 10 \times 10\,\text{mm}$ voxels that fill up the camera frustum. On YCB-V, this sampling step yields 230,383 query points, 0.85 s average image processing time and 66.8 AR (with 100 RANSAC iterations). Reducing the step size to 5 mm yields 1,852,690 points and 3.95 s, while only slightly improved accuracy of 67.0 AR. Enlarging the step size to 20 mm yields 28,232 points, improves the time to 0.63 s, and still achieves competitive accuracy of 66.2 AR. These results suggest that the method is relatively insensitive to the sampling density.

Number of Pose Fitting Iterations. We further investigate the effect of the number of RANSAC iterations on the accuracy and speed. On the YCB-V dataset, reducing the number of iterations from 200 to 50 and 10 decreases the AR score from 67.3 to 66.7 and 65.1, and improves the average processing time from 1.09 to 0.77 and 0.68 s, respectively. On the other hand, increasing the number of iterations from 200 to 500 yields the same AR score and higher average processing time of 1.67 s. Note that in the presented experiments we run both Kabsch-RANSAC and PnP-RANSAC algorithms for a fixed number of iterations. Further improvements in speed could be achieved by applying an early stopping criterion, which is typically based on the number of inliers *w.r.t.* the so-far-the-best pose hypothesis [16, 3].



(a) (b) (c) (d) (e)

Fig. 4. **Qualitative results on YCB-V and LM-O:** (a) An RGB input. (b) A mesh model reconstructed by Marching Cubes [45] from the signed distances predicted at 3D query points in the camera frustum. Mesh vertices are colored with the predicted 3D object coordinates. Note that the mesh is reconstructed only for visualization, not when estimating the object pose. (c) The reconstructed mesh from a novel view. (d) GT mesh colored with GT 3D object coordinates. (e) GT mesh in the view from (c).

6 Conclusion

We have proposed the first method for 6DoF object pose estimation based on a 3D implicit representation, which we call Neural Correspondence Field (NCF). The proposed method noticeably outperforms a baseline, which adopts a popular 2D-3D correspondence approach, and also all comparable methods on the YCB-V, LM-O, and LM datasets. Ablation studies and qualitative results demonstrate the ability of NCF to learn and incorporate priors about the whole object surface, which is important for handling challenging cases with occlusion.

References

1. Atzmon, M., Lipman, Y.: SAL: Sign agnostic learning of shapes from raw data. CVPR (2020) 3
2. Atzmon, M., Lipman, Y.: SALD: Sign agnostic learning with derivatives. ICLR (2021) 3
3. Baráth, D., Matas, J.: Graph-Cut RANSAC. CVPR (2018) 13
4. Baráth, D., Matas, J.: Progressive-X: Efficient, anytime, multi-model fitting algorithm. ICCV (2019) 8
5. Brachmann, E., Krull, A., Michel, F., Gumhold, S., Shotton, J., Rother, C.: Learning 6D object pose estimation using 3D object coordinates. ECCV (2014) 2, 3, 4, 5, 8, 9, 12
6. Brachmann, E., Michel, F., Krull, A., Yang, M.Y., Gumhold, S., Rother, C.: Uncertainty-driven 6D pose estimation of objects and scenes from a single RGB image. CVPR (2016) 4
7. Brunelli, R.: Template matching techniques in computer vision: Theory and practice. John Wiley & Sons (2009) 3
8. Chen, Z., Zhang, H.: Learning implicit fields for generative shape modeling. CVPR (2019) 3
9. Collet, A., Martinez, M., Srinivasa, S.S.: The MOPED framework: Object recognition and pose estimation for manipulation. IJRR (2011) 2, 3
10. Corona, E., Kundu, K., Fidler, S.: Pose estimation for objects with rotational symmetry. IROS (2018) 3
11. Curless, B., Levoy, M.: A volumetric method for building complex models from range images. SIGGRAPH (1996) 5
12. Deng, Y., Yang, J., Tong, X.: Deformed implicit field: Modeling 3d shapes with learned dense correspondence. CVPR (2021) 4
13. Denninger, M., Sundermeyer, M., Winkelbauer, D., Olefir, D., Hodaň, T., Zidan, Y., Elbadrawy, M., Knauer, M., Katam, H., Lodhi, A.: BlenderProc: reducing the reality gap with photorealistic rendering. RSS Workshops (2020) 9
14. Denninger, M., Sundermeyer, M., Winkelbauer, D., Zidan, Y., Olefir, D., Elbadrawy, M., Lodhi, A., Katam, H.: BlenderProc. arXiv preprint arXiv:1911.01911 (2019) 9
15. Drost, B., Ulrich, M., Navab, N., Ilic, S.: Model globally, match locally: Efficient and robust 3D object recognition. CVPR (2010) 3, 11
16. Fischler, M.A., Bolles, R.C.: Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM (1981) 2, 5, 7, 12, 13
17. Gropp, A., Yariv, L., Haim, N., Atzmon, M., Lipman, Y.: Implicit geometric regularization for learning shapes. ICML (2020) 3
18. Güler, R.A., Neverova, N., Kokkinos, I.: Densepose: Dense human pose estimation in the wild. CVPR (2018) 4
19. Guo, Y., Bennamoun, M., Sohel, F., Lu, M., Wan, J., Kwok, N.M.: A comprehensive performance evaluation of 3D local feature descriptors. IJCV (2016) 3
20. He, Y., Huang, H., Fan, H., Chen, Q., Sun, J.: Ffb6d: A full flow bidirectional fusion network for 6d pose estimation. CVPR (2021) 11
21. Hinterstoisser, S., Lepetit, V., Ilic, S., Holzer, S., Bradski, G., Konolige, K., Navab, N.: Model based training, detection and pose estimation of texture-less 3D objects in heavily cluttered scenes. ACCV (2012) 3, 9, 11, 12

22. Hinterstoisser, S., Lepetit, V., Rajkumar, N., Konolige, K.: Going further with point pair features. ECCV (2016) 3
23. Hodaň, T., Baráth, D., Matas, J.: EPOS: Estimating 6D pose of objects with symmetries. CVPR (2020) 2, 3, 4, 8, 10, 11
24. Hodaň, T., Michel, F., Brachmann, E., Kehl, W., Glent Buch, A., Kraft, D., Drost, B., Vidal, J., Ihrke, S., Zabulis, X., Sahin, C., Manhardt, F., Tombari, F., Kim, T.K., Matas, J., Rother, C.: BOP: Benchmark for 6D object pose estimation. ECCV (2018) 1, 3
25. Hodaň, T., Sundermeyer, M., Drost, B., Labbé, Y., Brachmann, E., Michel, F., Rother, C., Matas, J.: BOP challenge 2020 on 6D object localization. ECCVW (2020) 1, 3, 8, 9, 11, 12, 13
26. Hodaň, T., Zabulis, X., Lourakis, M., Obdržálek, Š., Matas, J.: Detection and fine 3D pose estimation of texture-less objects in RGB-D images. IROS (2015) 3
27. Hu, Y., Hugonot, J., Fua, P., Salzmann, M.: Segmentation-driven 6D object pose estimation. CVPR (2019) 3
28. Huang, Z., Xu, Y., Lassner, C., Li, H., Tung, T.: Arch: Animatable reconstruction of clothed humans. CVPR (2020) 1
29. Huber, P.J.: Robust estimation of a location parameter. Breakthroughs in statistics (1992) 7
30. Jafari, O.H., Mustikovela, S.K., Pertsch, K., Brachmann, E., Rother, C.: iPose: Instance-aware 6D pose estimation of partly occluded objects. ACCV (2018) 3, 4
31. Kabsch, W.: A discussion of the solution for the best rotation to relate two sets of vectors. Acta Crystallographica Section A (1978) 2, 5
32. Karunratanakul, K., Yang, J., Zhang, Y., Black, M.J., Muandet, K., Tang, S.: Grasping field: Learning implicit representations for human grasps. 3DV (2020) 4
33. Kehl, W., Manhardt, F., Tombari, F., Ilic, S., Navab, N.: SSD-6D: Making RGB-based 3D detection and 6D pose estimation great again. ICCV (2017) 3
34. Koenig, R., Drost, B.: A hybrid approach for 6dof pose estimation. ECCVW (2020) 11
35. Krull, A., Brachmann, E., Michel, F., Ying Yang, M., Gumhold, S., Rother, C.: Learning analysis-by-synthesis for 6D pose estimation in RGB-D images. ICCV (2015) 3
36. Kulkarni, N., Gupta, A., Fouhey, D.F., Tulsiani, S.: Articulation-aware canonical surface mapping. CVPR (2020) 4
37. Kulkarni, N., Johnson, J., Fouhey, D.F.: What's behind the couch? directed ray distance functions (drdf) for 3d scene reconstruction. arXiv e-prints (2021) 1
38. Labbé, Y., Carpentier, J., Aubry, M., Sivic, J.: CosyPose: consistent multi-view multi-object 6D pose estimation. ECCV (2020) 3, 10, 11
39. Lepetit, V., Moreno-Noguer, F., Fua, P.: EPnP: An accurate O(n) solution to the PnP problem. IJCV (2009) 2, 8
40. Li, C., Bai, J., Hager, G.D.: A unified framework for multi-view multi-class object pose estimation. ECCV (2018) 3
41. Li, Z., Wang, G., Ji, X.: CDPN: Coordinates-based disentangled pose network for real-time RGB-based 6-DoF object pose estimation. ICCV (2019) 3, 4, 8, 10, 11
42. Liu, F., Tran, L., Liu, X.: Fully understanding generic objects: Modeling, segmentation, and reconstruction. CVPR (2021) 3
43. Liu, J., Zou, Z., Ye, X., Tan, X., Ding, E., Xu, F., Yu, X.: Leaping from 2D detection to efficient 6DoF object pose estimation. ECCVW (2020) 10, 11
44. Lombardi, S., Simon, T., Saragih, J., Schwartz, G., Lehrmann, A., Sheikh, Y.: Neural volumes: Learning dynamic renderable volumes from images. TOG (2019) 4

45. Lorensen, W.E., Cline, H.E.: Marching cubes: A high resolution 3d surface construction algorithm. SIGGRAPH (1987) 2, 14
46. Lowe, D.G., et al.: Object recognition from local scale-invariant features. ICCV (1999) 3
47. Manhardt, F., Arroyo, D.M., Rupprecht, C., Busam, B., Navab, N., Tombari, F.: Explaining the ambiguity of object detection and 6D pose from visual data. ICCV (2019) 3
48. Mescheder, L., Oechsle, M., Niemeyer, M., Nowozin, S., Geiger, A.: Occupancy networks: Learning 3d reconstruction in function space. CVPR (2019) 3, 4
49. Michel, F., Alexander Kirillov, Brachmann, E., Krull, A., Gumhold, S., Savchynskyy, B., Rother, C.: Global hypothesis generation for 6D object pose estimation. CVPR (2017) 4
50. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: NeRF: Representing scenes as neural radiance fields for view synthesis. ECCV (2020) 4
51. Neverova, N., Novotny, D., Khalidov, V., Szafraniec, M., Labatut, P., Vedaldi, A.: Continuous surface embeddings. NeurIPS (2020) 4
52. Newell, A., Yang, K., Deng, J.: Stacked hourglass networks for human pose estimation. ECCV (2016) 4
53. Niemeyer, M., Mescheder, L., Oechsle, M., Geiger, A.: Differentiable Volumetric Rendering: Learning Implicit 3D Representations without 3D Supervision. CVPR (2020) 4
54. Oberweger, M., Rad, M., Lepetit, V.: Making deep heatmaps robust to partial occlusions for 3D object pose estimation. ECCV (2018) 3
55. Oechsle, M., Mescheder, L., Niemeyer, M., Strauss, T., Geiger, A.: Texture fields: Learning texture representations in function space. ICCV (2019) 4
56. Palafox, P., Božič, A., Thies, J., Nießner, M., Dai, A.: Npms: Neural parametric models for 3d deformable shapes. ICCV (2021) 4
57. Park, J.J., Florence, P., Straub, J., Newcombe, R., Lovegrove, S.: Deepsdf: Learning continuous signed distance functions for shape representation. CVPR (2019) 3, 5, 6, 7, 10
58. Park, K., Sinha, U., Barron, J.T., Bouaziz, S., Goldman, D.B., Seitz, S.M., Martin-Brualla, R.: Nerfies: Deformable neural radiance fields. ICCV (2021) 4
59. Park, K., Patten, T., Vincze, M.: Pix2Pose: Pixel-wise coordinate regression of objects for 6D pose estimation. ICCV (2019) 2, 3, 4, 8, 10, 11
60. Pavlakos, G., Zhou, X., Chan, A., Derpanis, K.G., Daniilidis, K.: 6-DoF object pose from semantic keypoints. ICRA (2017) 3
61. Peng, S., Liu, Y., Huang, Q., Zhou, X., Bao, H.: PVNet: Pixel-wise voting network for 6DoF pose estimation. CVPR (2019) 2, 3
62. Pumarola, A., Corona, E., Pons-Moll, G., Moreno-Noguer, F.: D-NeRF: Neural Radiance Fields for Dynamic Scenes. CVPR (2020) 4
63. Rad, M., Lepetit, V.: BB8: A scalable, accurate, robust to partial occlusion method for predicting the 3D poses of challenging objects without using depth. ICCV (2017) 2, 3
64. Raposo, C., Barreto, J.P.: Using 2 point+normal sets for fast registration of point clouds with small overlap. ICRA (2017) 11
65. Rodrigues, P., Antunes, M., Raposo, C., Marques, P., Fonseca, F., Barreto, J.: Deep segmentation leverages geometric pose estimation in computer-aided total knee arthroplasty. Healthcare Technology Letters (2019) 11

66. Saito, S., Huang, Z., Natsume, R., Morishima, S., Kanazawa, A., Li, H.: Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. ICCV (2019) 1, 2, 4, 5, 6, 10
67. Saito, S., Simon, T., Saragih, J., Joo, H.: Pifuhd: Multi-level pixel-aligned implicit function for high-resolution 3d human digitization. CVPR (2020) 1, 4
68. Shotton, J., Glocker, B., Zach, C., Izadi, S., Criminisi, A., Fitzgibbon, A.: Scene coordinate regression forests for camera relocalization in rgb-d images. CVPR (2013) 4
69. Sitzmann, V., Chan, E., Tucker, R., Snavely, N., Wetzstein, G.: Metasdf: Meta-learning signed distance functions. NeurIPS (2020) 3
70. Sitzmann, V., Zollhöfer, M., Wetzstein, G.: Scene representation networks: Continuous 3d-structure-aware neural scene representations. NeurIPS (2019) 4
71. Sundermeyer, M., Marton, Z.C., Durner, M., Triebel, R.: Augmented autoencoders: Implicit 3D orientation learning for 6D object detection. IJCV (2019) 3, 11
72. Tejani, A., Tang, D., Kouskouridas, R., Kim, T.K.: Latent-class hough forests for 3D object detection and pose estimation. ECCV (2014) 3
73. Tekin, B., Sinha, S.N., Fua, P.: Real-time seamless single shot 6D object pose prediction. CVPR (2018) 2, 3
74. Tewari, A., Thies, J., Mildenhall, B., Srinivasan, P., Tretschk, E., Yifan, W., Lassner, C., Sitzmann, V., Martin-Brualla, R., Lombardi, S., et al.: Advances in neural rendering. Computer Graphics Forum (2022) 4
75. Tieleman, T., Hinton, G.: Rmsprop: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural Networks for Machine Learning (2012) 10
76. Tremblay, J., To, T., Sundaralingam, B., Xiang, Y., Fox, D., Birchfield, S.: Deep object pose estimation for semantic robotic grasping of household objects. CoRL (2018) 3
77. Tretschk, E., Tewari, A., Golyanik, V., Zollhöfer, M., Lassner, C., Theobalt, C.: Non-rigid neural radiance fields: Reconstruction and novel view synthesis of a dynamic scene from monocular video. ICCV (2021) 4
78. Vidal, J., Lin, C.Y., Lladó, X., Martí, R.: A method for 6D pose estimation of free-form rigid objects using point pair features on range data. Sensors (2018) 3, 11
79. 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. CVPR (2019) 3, 4, 7, 8
80. Xiang, Y., Schmidt, T., Narayanan, V., Fox, D.: PoseCNN: A convolutional neural network for 6D object pose estimation in cluttered scenes. RSS (2018) 3, 9, 12
81. Xu, Q., Wang, W., Ceylan, D., Mech, R., Neumann, U.: Disn: Deep implicit surface network for high-quality single-view 3d reconstruction. Advances in Neural Information Processing Systems (2019) 4
82. Yariv, L., Kasten, Y., Moran, D., Galun, M., Atzmon, M., Ronen, B., Lipman, Y.: Multiview neural surface reconstruction by disentangling geometry and appearance. NeurIPS (2020) 4
83. Zakharov, S., Shugurov, I., Ilic, S.: DPOD: 6D pose object detector and refiner. ICCV (2019) 2, 3, 4, 8
84. Zheng, Z., Yu, T., Liu, Y., Dai, Q.: Pamir: Parametric model-conditioned implicit representation for image-based human reconstruction. TPAMI (2021) 1