6DoF Pose Estimation of Transparent Object from a Single RGB-D Image

Chi Xu 1,2,3,†, Jiale Chen 1,2,*,†, Mengyang Yao 1,2,†, Jun Zhou 1,2,†, Lijun Zhang 1,2,† and Yi Liu 4,5

1 School of Automation, China University of Geosciences, Wuhan 430074, China; xuchi@cug.edu.cn (C.X.); vtasyao@163.com (M.Y.); jchow@cug.edu.cn (J.Z.); lijunzh@cug.edu.cn (L.Z.)
2 Hubei Key Laboratory of Advanced Control and Intelligent Automation for Complex Systems, Wuhan 430074, China
3 Engineering Research Center of Intelligent Technology for Geo-Exploration, Ministry of Education, Wuhan 430074, China
4 CRRC Zhuzhou Electric Locomotive Co., Ltd. 1 TianXin Road; Zhuzhou 412000, China; liuyi_hust@163.com
5 National Innovation Center of Advanced Rail Transit Equipment, Zhuzhou 412000, China
*
† These authors contributed equally to this work.

Abstract: 6DoF object pose estimation is a foundation for many important applications, such as robotic grasping, automatic driving, and so on. However, it is very challenging to estimate 6DoF pose of transparent object which is commonly seen in our daily life, because the optical characteristics of transparent material lead to significant depth error which results in false estimation. To solve this problem, a two-stage approach is proposed to estimate 6DoF pose of transparent object from a single RGB-D image. In the first stage, the influence of the depth error is eliminated by transparent segmentation, surface normal recovering, and RANSAC plane estimation. In the second stage, an extended point-cloud representation is presented to accurately and efficiently estimate object pose. As far as we know, it is the first deep learning based approach which focuses on 6DoF pose estimation of transparent objects from a single RGB-D image. Experimental results show that the proposed approach can effectively estimate 6DoF pose of transparent object, and it out-performs the state-of-the-art baselines by a large margin.

Keywords: 6DoF pose estimation; transparent object; human-computer interaction

1. Introduction

6DoF (Degrees of Freedom) pose estimation aims at estimating an object’s rotation (3DoF) and translation (3DoF) in the camera coordinate frame [1–3]. In some papers, “6DoF” is also referred to as “6D” for short. It is a key technology closely related to many important real-world applications, such as robotic grasping [4,5], automatic driving [6,7], augmented reality [8,9], and so on. With the emergence of consumer-level RGB-D sensor (e.g., Kinect, Intel RealSense, etc.), 6DoF object pose estimation accuracy has been significantly boosted by using RGB-D image [2,3,10,11].

Transparent objects (e.g., glasses, plastic bottles, bowls, etc.) are commonly seen in our daily environments, such as kitchen, office, living room, canteen, and so on. However, existing pose estimation methods for ordinary objects cannot deal with transparent ones correctly (please refer to Figure 1), because the optical characteristics of transparent material lead to significant depth error [12] in D-channel of RGB-D image (the D-channel encodes the depth from the object to the camera, which contains important hint to retrieve the 3D geometric information of the observed
object). As can be seen in Figure 2, the depth error of transparent object can be classified into two types: (i) missing depth, i.e. the depth of specific regions is missing due to specular reflection on surface of transparent object; and (ii) background depth, i.e., instead of the true depth on object surface, the false distorted depth on background behind the object is captured, since light passes through the transparent material and refraction occurs. The cause of the depth error is illustrated in Figure 3. The above-mentioned depth errors distort the 3D geometric information of the transparent object observed, and it significantly degrades the pose estimation accuracy of transparent object.

![Figure 1: Pose estimation results of transparent objects. (a) DenseFusion [2], (b) Our proposed approach. The object pose is expressed as tight oriented bounding box of the object model. DenseFusion [2] is one of the state-of-the-art methods for 6DoF object pose estimation from a single RGB-D image.](image1)

![Figure 2: Depth errors of transparent objects: (a) RGB-channel of non-transparent objects; (b) D-channel of non-transparent objects; (c) RGB-channel of transparent objects; and (d) D-channel of transparent objects. For the transparent objects, there exist two types of depth error in the D-channel. The depth error dramatically distorts the 3D geometric information of transparent objects observed.](image2)
In this paper, we propose an accurate and efficient approach for 6DoF pose estimation of transparent object from a single RGB-D image. As illustrated in Figure 4, the proposed approach contains two stages: In the first stage, we eliminate the influence of depth errors by transparent segmentation, surface normal recovering, and RANSAC plane estimation. In the second stage, an extended point cloud representation is constructed based on the output of the first stage, and the color feature is extracted from cropped color patch. The extended point-cloud and the color feature extracted are then fed into a DenseFusion-like network structure [2] for 6DoF pose estimation. The block-diagram of our approach is shown in Figure 5. The RGB-channel is fed into the transparent segmentation module to retrieve bounding box and mask of transparent object. Then, the color feature is extracted, the surface normal is recovered, and the plane is estimated. By taking the normal, the plane, and the UV map as inputs, the extended point-cloud is constructed. Finally, based on the color feature and the extended point-cloud, the 6DoF object pose is estimated.

The recovered surface normal, the plane, and the UV map are essential components of the extended point-cloud representation, which contains rich geometric information for 6DoF pose estimation: (1) the recovered surface normal contains important hint to estimate the object’s relative pose (3DoF rotation); (2) the plane where the object placed is closely related to the object’s 3D position; and (3) the UV map encodes the 2D coordinates of points on image, which is crucial for 6DoF object pose estimation [1,13].

The work most related to our approach is ClearGrasp [12]. Similar to Sajjan et al. [12], we estimate surface normal of transparent object. Different from Sajjan et al. [12], we focus on 6DoF pose estimation, while ClearGrasp aims at depth reconstruction.

Our approach is very different from the direct feeding of the reconstructed depth into an ordinary RGB-D pose estimator, as in [12]. The differences are mainly in the following to two aspects: (1) ClearGrasp [12] reconstructs depth using a global optimization scheme which is time consuming, but we do not reconstruct depth so that the inferring efficiency is largely accelerated. (2) The ordinary RGB-D pose estimator takes the depth as input, and the depth is converted into classic point-cloud for pose estimation. However, our approach does not rely on classic point-cloud, therefore depth reconstruction is not required.

Overall, the contributions of this paper are twofold:

- We propose a new deep learning based approach which focuses on 6DoF pose estimation of transparent object from a single RGB-D image. Discriminative high-level deep features are retrieved through a two-stage end-to-end neural network, which result in an accurate 6DoF pose estimation.
- We introduce a novel extended point cloud representation for 6DoF pose estimation. Different from classic point cloud, the representation does not require depth as input.
With this representation, object pose can be efficiently recovered without the time consuming depth reconstruction.

Experimental results show that the proposed approach significantly outperforms state-of-the-art baselines in terms of accuracy and efficiency.

Figure 4. Framework of our proposed approach. In the RANSAC Plane Estimation step, the noisy depth in the segmented transparent region is removed before plane estimation. After the Color Feature Extraction step, 500 pixels are randomly sampled in the segmented transparent region.

Figure 5. Block-diagram of our proposed approach.

2. Related Work

This paper focuses on 6DoF pose estimation of transparent object from a single RGB-D image. In this section, we review the related literature regarding the following aspects:

**Traditional object pose estimation.** Traditional methods primarily utilize hand-crafted low-level features for 6DoF object pose estimation. In [14,15], oriented point pair is used to describe the global information of object model. In [16], a template matching method is proposed to detect 3D objects. In [17], a RANSAC-based scheme is designed to randomly match three correspondences between the scene and the object model. As the traditional methods rely on low-level features, they are not as accurate as deep learning based methods.

**Deep learning based object pose estimation.** In recent years, deep learning has largely improved the accuracy and robustness of 6DoF object pose estimation, as high-level deep features are much
more discriminative than low-level traditional features. Early works \cite{10,11} directly apply 2D CNN on RGB-D image for 6DoF pose regression, while 2D CNN does not characterize 3D geometric information well. To better explore the 3D geometric information, the 3D space corresponding to depth image is divided into voxel grids, and then 3D CNN is applied on voxels \cite{18–20}. Higher-dimensional convolution on 3D voxel requires huge computational resources. To improve the computational efficiency, point-net based methods \cite{21,22} directly extract deep geometric features from classic point-cloud while retaining the computational efficiency. DenseFusion \cite{2} further augments the point-cloud based geometric features by embedding the corresponding color features, and it significantly improves the pose estimation accuracy. Based on the geometric and color features of Wang et al. \cite{2}, Rotation Anchor \cite{3} uses a discrete-continuous formulation for rotation regression to resolve local-optimum problem of symmetric objects. Nevertheless, the above-mentioned ordinary pose estimators cannot correctly deal with transparent object.

Detection of transparent objects. Transparent object is a very difficult topic in computer vision field. The appearance of transparent objects can vary dramatically because of the reflective and refractive nature of transparent material. Many research works focus on transparent object detection. Fritz et al. \cite{23} proposed an additive latent feature for transparent object recognition. McHenry et al. \cite{24} identified glass edges by training hierarchical SVM. Philips et al. \cite{25} segregated semi-transparent objects by a stereoscopic cue. Xie et al. \cite{26} proposed a boundary-aware approach to segment transparent object. In \cite{27}, glass regions are segmented using geodesic active contour based optimization. In \cite{28}, glass objects are localized by joint inferring boundary and depth. In \cite{29–31}, transparent objects are detected using CNN networks. However, these methods only detect transparent object, and do not estimate 6DoF object pose.

3D reconstruction of transparent objects. Depth can be reconstructed from constrained background environments. In \cite{32,33}, transparent objects are reconstructed with known background patterns. In \cite{34–36}, the depth of transparent objects is reconstructed using time of flight camera, since glass absorbs light of certain wavelengths. 3D geometry can be reconstructed from multiple views or 3D scanning. Ji et al. \cite{37} conducted a volumetric reconstruction of transparent objects by fusing depth and silhouette from multiple images with known poses. Li et al. \cite{38} presented a physically based network to recover 3D shape of transparent object from multiple color images. Albrecht et al. \cite{39} reconstructed the geometry of transparent object from point cloud of multiple views. Different from the above works, Sajjan et al. \cite{12} reconstructed the depth as well as the surface normal of transparent object from a single RGB-D image. However, the 6DoF pose estimation is still not addressed in \cite{12}.

Pose estimation of transparent objects. Estimating pose of transparent object from a single RGB-D image is a challenging task. Some works estimate pose using traditional features. In \cite{40,41}, the pose of a rigid transparent object is estimated by 2D edge feature analysis. In \cite{42}, SIFT features are used to recognize the transparent object. However, low-level traditional features are not as discriminative as high-level deep features. To the best of our knowledge, in previous work, high-level deep features have not been utilized to estimate 6DoF transparent object pose from a single RGB-D image. It is worth noting that other sensors can also be used for transparent object pose estimation. For example, transparent object pose can be estimated through a monocular color camera \cite{43,44}, but the translation estimation along the z-axis tends to be inaccurate due to lack of 3D depth information. Stereo camera \cite{45,46}, light field camera \cite{47}, single pixel camera \cite{48}, and microscope–camera system \cite{49} can be used for object pose estimation, but these works are very different from this paper and are not discussed further.

3. Method

This work aims to predict the 6DoF pose of known transparent object in the camera coordinate frame using a single RGB-D image. The object pose is represented by a 3D rigid transformation \((R, T)\) with rotation \(R \in SO(3)\) and translation \(T \in R^3\). Normally, there exist multiple object instances
within one image. For each segmented object instance, the proposed approach estimates the object pose through a two-stage framework, as shown in Figure 4. In the first stage, there are three modules: transparent segmentation, surface normal estimation, and RANSAC plane estimation. The results of the first stage are fed into the second stage for further processing. In the second stage, there are three modules: extended point-cloud, color feature extraction, and 6DoF pose estimation. These six modules are the key subsections of our approach. Details are described as follows.

3.1. The First Stage

The first stage takes a single RGB-D image as input. The RGB-D image contains two parts: RGB-channel and D-channel. The RGB-channel is used to segment the 2D transparent region and recover the 3D surface normal. The D-channel is used to estimate the 3D plane where the object is placed. The roles of the modules are as follow: the transparent segmentation module identifies the transparent region and segments the object instances; the surface normal estimation module and the RANSAC plane estimation module together recover the transparent object’s 3D geometric information.

**Transparent segmentation.** The transparent object instance is identified by the transparent segmentation module. We segment the region of interest corresponding to transparent object instance through a Mask R-CNN [50] network. Given a single RGB-D image, its RGB-channel is fed into the segmentation network as input, and the output is a list of detected object instances and the corresponding segmentation maps. The transparent region is segmented for three reasons: (1) to remove the noisy depth of transparent region and keep the reliable depth of non-transparent region; (2) to calculate 2D bounding box and crop image patches for further feature extraction; and (3) to sample extended point-cloud on transparent object.

Mask R-CNN [50] is an accurate and mature component of many 6DoF pose estimation methods, such as DenseFusion [2] and Rotation Anchor [3]. In this paper, the Mask R-CNN component is trained and evaluated using the standard scheme, as in previous works [2,3,50]. The number of classes is 5 in our experiments. Mask R-CNN is very accurate for transparent segmentation. To investigate how much the performance of Mask R-CNN impacts the rest of the pipeline, we evaluate the whole pipeline using the Mask R-CNN segmentation and the ground-truth segmentation, respectively. The results show that the 6DoF pose estimation accuracy based on the ground-truth segmentation is only 0.7% higher than that based on the Mask R-CNN segmentation. Besides, other instance segmentation networks (such as SegNet [51] and RefineNet [52]) can also be adopted for transparent segmentation.

**Surface normal recovering.** Our surface normal estimation network adopts an encoding–decoding structure which is the same as that of Sajjan et al. [12]. The network structure is shown in Figure 6. Firstly, taking the RGB-channel as input, one convolution layer and several residual blocks are used to obtain low-level features. Secondly, the Atrous Spatial Pyramid Pooling [53] sub-network is used to sample high-level features. It not only captures the context in multi-scale but also retains the global characteristics of feature through average pooling. We apply skip connections between the low-level features and the high-level features to ensure the integrity. Beside, we use many residual blocks with dilated convolution, which increases the receptive field of individual neurons while maintaining the resolution of the output feature map. Thirdly, L2 normalization is applied on the three-channel output of the network, so that the output on each pixel is enforced to be a unit vector, which represents the estimated normal. We calculate the cosine similarity between the estimated normal and ground-truth normal as follows:

\[
L_{\text{norm}} = \frac{1}{k} \sum_{i \in K} \cos(N_i - \tilde{N}_i),
\]

in which \(L_{\text{norm}}\) denotes the normal estimation loss, \(K\) denotes the pixel set within the image, and \(k\) denotes the number of pixels in \(K\). \(N_i\) and \(\tilde{N}_i\) denote the estimated normal and the ground-truth on \(i\)th pixel, respectively.
RANSAC plane estimation. We estimate the 3D plane where the transparent object is placed from the D-channel. We preprocess the original depth using the output of transparent segmentation. The depth of transparent region is discarded to eliminate the influence of depth error, and the depth of non-transparent region is retained for plane estimation. With the camera’s intrinsic parameter matrix, the depth is converted into 3D point cloud. We estimate the 3D plane where the transparent object is placed by a RANSAC (Random Sample Consensus) plane detection algorithm [54]. The detected plane is considered to be valid if the number of inlier points is more than a specific threshold. To ensure the robustness of plane estimation, we repeat the RANSAC plane fitting with rest points, and then the depth of the best fitted plane is selected to be fed into the second stage.

3.2. The Second Stage

The second stage takes the results of the first stage as input and then estimates the 6DoF object pose. Firstly, the color feature is extracted from cropped color patch. Secondly, the UV code, the surface normal, and the depth value on plane are concatenated to form a data structure named “extended point-cloud”. Thirdly, based on the extended point-cloud and the color feature, the 6DoF object pose is estimated by a DenseFusion-like network [2].

Color Feature Extraction. For each segmented object instance, we crop color patch $P_{color}$ from the RGB-channel. $P_{color}$ is resized to an uniform size of $80 \times 80$ for color feature extraction. The color feature extraction network is a CNN-based encoder–decoder architecture. It takes $P_{color}$ as input and outputs color feature with the same size as $P_{color}$. The dimensional of the output color feature is 32. We randomly sample 500 pixels within the segmented transparent region. For each sampled pixel $x$ in the patch, the color feature on that pixel is $P_{color}(x)$. It is later fed into the DenseFusion-like network for 6DoF pose estimation.

Extended point-cloud. Ordinary methods [2,3] take classic point-cloud of depth as input for pose estimation. However, we do not use the point-cloud of original depth as input, as it has been dramatically distorted by depth error. We also do not reconstruct depth to obtain rectified point-cloud, because accurate depth reconstruction is time consuming. We rectify the distorted geometry by surface normal recovery. Since surface normal alone is insufficient for absolute pose estimation [12], an extended point-cloud representation is proposed to estimate 6DoF object pose in our research.

For each segmented transparent object instance, we crop patches from the UV encoding map, the estimated surface normal, and the depth of the estimated plane. The UV encoding map encodes the 2D $(u, v)$ coordinates of each pixel on image. The estimated surface normal and depth of the estimated plane are the output of the first stage. The cropped patches $P_{UV}, P_{norm}, P_{plane}$, are resized to an uniform size of $80 \times 80$ for extended point-cloud construction.
For each sampled pixel $x$ in the patch, the extended point-cloud is defined by concatenating $P_{UV}(x)$, $P_{norm}(x)$, and $P_{plane}(x)$. The roles of these three components are as follows: (1) $P_{UV}(x)$ indicates the 2D location of a pixel on image plane, and it is an important hint for 6DoF pose estimation [1,13]; (2) $P_{norm}(x)$ indicates the surface normal of object, and it is important for relative pose (3DoF rotation) estimation; and (3) $P_{plane}(x)$ provides the hint where the object is placed, and it helps conduct more accurate 3DoF translation estimation.

**6DoF Pose estimation.** Taking the extended point-cloud and the color feature as input, we adopt a DenseFusion-like network structure (similar to [2,3]) for 6DoF pose estimation. As illustrated in Figure 7, edge convolution [55] is applied on the extended point-cloud to extract geometry features. Then, the geometry and color features are densely concatenated for 6DoF pose estimation. We randomly sample $N$ pixels for feature extraction. In this work, the value of $N$ is 500, the same as that of Rotation Anchor [3] and DenseFusion [2]. The number of anchors is 60, the same as that of Rotation Anchor [3].

![Figure 7. Pipeline of the 6DoF pose estimation network.](image)

The network structure of ours is very similar to that of Rotation Anchor [3], and the difference is that Rotation Anchor takes 3D classic point-cloud as input but ours takes 6D extended point-cloud as input. Thus, the weight matrix of the first fully connected layer in the geometry feature branch is $6 \times 64$, and that of Rotation Anchor is $3 \times 64$. The pose estimation process is described as follows.

Instead of directly regressing the translation [56,57] or key-points [58,59] of object, predicting vectors that represent the direction from pixel toward object is more robust [3]. For each of the pixel, we predict the vector points to the center of the object and normalize it to a unit vector. This vector-field representation focuses on local features and is sensitive to occlusion and truncation. We use the unit vectors to generate coordinates of object center in a RANSAC-based voting scheme. With the coordinates of the center point, we can obtain the 3D translation of the object.

Due to pose ambiguity caused by symmetrical objects, directly regressing rotation is always a challenge for 6DoF pose estimation task. Early experimentation [11,60] shows clearly that using discrete-continuous regression scheme is effective to obtain accurate rotation. However, SSD-6D [60] requires prior knowledge of 3D models to manually select classified viewpoints, and Li et al. [11] did not enforce local estimation for every rotation classification. Similar to Tian et al. [3], rotation anchors are used to represent sub-part of the whole rotation space $SO(3)$ which is divided equally. For each rotation anchor $\hat{R}_j$, where the subscript $j$ denotes the index of the anchor, we predict the rotation offset $\Delta R_j$ and the predicted rotation:

$$R_j = \Delta R_j \hat{R}_j.$$  

(2)
Additionally, Confidence $C_j$ is predicted to represent the similarity between $R_j$ and the ground truth. The prediction of the anchor with the highest confidence is selected as the output. The anchor index is selected as $\hat{j} = \arg \max_j C_j$; the output 3DoF rotation $R = R_{\hat{j}}$ and confidence $C = C_{\hat{j}}$.

The loss $L_{\text{pose}}$ aims to constrain distance between the estimated pose and the ground truth pose. It is divided into three parts:

$$L_{\text{pose}} = \lambda_3 L_{\text{shape}} + \lambda_4 L_{\text{reg}} + \lambda_5 L_{\text{t}}, \quad (3)$$

where $L_{\text{shape}}$ is an extension of ShapeMatch-Loss [56] in the aspect of difference in object diameter. $L_{\text{shape}}$ normalize the loss with object diameter:

$$L_{\text{shape}} = \sum_{x_1 \in M} \min_{x_2 \in M} \frac{|| R x_1 - \tilde{R} x_2 ||_2}{m \times d \times C} + \log C, \quad (4)$$

where $m$ denotes the number of points in 3D object model $M$, $d$ denotes the diameter of object, $\tilde{R}$ denotes the ground truth rotation, and $C$ denotes the confidence. The loss $L_{\text{reg}}$ constrains the regularization range of $\Delta R_j$,

$$L_{\text{reg}} = \sum_{j} \max(0, \max_{j \neq k} < q_j, \hat{q}_k > - < q_j, \hat{q}_j >), \quad (5)$$

where $q_j$ and $\hat{q}_j$ are quaternion representations of $R_j$ and $\hat{R}_j$. The closer the estimated rotation is to the ground truth rotation, the larger the dot product of the two quaternion matrices becomes. The loss $L_{\text{t}}$ constrains the translation of object, and it is calculated as the smooth L1 norm distance between the RANSAC-based voting results and the ground-truth translation.

### 3.3. Dataset

We evaluated our approach on ClearGrasp dataset [12] which contains five common transparent objects with symmetric properties. Occlusions and truncation exist in the images captured, which make this dataset challenging. As far as we know, it is the only publicly available RGB-D dataset applicable for transparent object 6DoF pose estimation. ClearGrasp dataset contains synthetic images and real images. For the quantitative experiments (Sections 4.4–4.6), we used the synthetic data in which the ground-truth pose is available. For the qualitative experiments (Section 4.6), we used the real images and synthetic images. From the synthetic data, we randomly picked 6000 images for training and 1500 images for testing. Each image contains multiple annotated objects. For the training set, there are 14,716 instances, and, for the testing set, there are 4765 instances. All the compared methods were trained and tested using the same training and testing split. DenseFusion [2] and Rotation Anchor [3] were trained and tested using the depth generated by the compared depth completion methods. For initialization, we used the pretrained model publicly available, and then fine-tuned the network using the training data. In the experiments, we adopted the standard depth completion pipeline of ClearGrasp [12].

### 4. Experiments

#### 4.1. Experimental Settings

All experiments were performed on a computer with an Intel Xeon Gold 6128 CPU and a NVIDIA TITAN V GPU. The proposed approach was implemented with PyTorch [61]. The network was trained using an Adam optimizer [62] with an initial learning rate of 0.001. The batch size was set to 4, and the images were resized to a resolution of 640 × 480. We adopted intermediate supervision to solve vanishing gradients problem which calculates losses at different stages of the network. The total loss is $L = \lambda_1 L_{\text{norm}} + \lambda_2 L_{\text{pose}}$, where $L_{\text{norm}}$ is used to constrain the surface normal estimation and $L_{\text{pose}}$ is used to constrain the final 6DoF pose estimation. We set the parameter $\lambda_1$ to decrease from 1 to
0, and then freeze the weights of normal estimation sub-network, while we set the parameter $\lambda_2$ to increase from 0 to 1. $L_{\text{pose}} = \lambda_3 L_{\text{shape}} + \lambda_4 L_{\text{reg}} + \lambda_5 L_t$. Following Tian et al. [3], $\lambda_3$, $\lambda_4$, and $\lambda_5$ were set as 1, 2, and 5, respectively. The training stopped when the number of epochs reached 80.

4.2. Evaluation Metric

Typically, ADD metric [63] is used to evaluate pose error for asymmetric objects, and ADD-S metric [56] is used for both symmetric and asymmetric objects. Given the ground truth pose $(R, T)$ and the estimated pose $(\tilde{R}, \tilde{T})$, ADD metric is defined as the average distance between the corresponding points after applying the transformations to the 3D model:

$$ADD = \frac{1}{m} \sum_{x \in M} || (Rx + T) - (\tilde{Rx} + \tilde{T}) ||,$$

where $x$ denotes a point within the 3D object model $M$ and $m$ denotes the number of points within the model.

For symmetric object which has multiple visually correct poses, ADD will cause many misjudgments. As only one of the correct poses is labeled as ground-truth, the estimation is considered as correct only if it matches with the ground-truth pose. However, other visually correct poses will be judged as false estimations. To resolve this problem, the overall evaluation of both symmetric and asymmetric objects are taken into account by ADD-S metric. ADD-S is defined as the average distance to the closest model point:

$$ADD-S = \frac{1}{m} \sum_{x_1 \in M} \min_{x_2 \in M} || (Rx_1 + T) - (\tilde{Rx}_2 + \tilde{T}) ||.$$

In our experiments, we use ADD-S metric for evaluation, because transparent object are commonly symmetric. The estimated pose is considered as correct if ADD-S is less than a given threshold. We set the threshold to 10% of the object diameter, the same as that of many previous works [2,3].

4.3. Accuracy

We compared our approach with state-of-the-art baselines. Generally, to eliminate the depth error of transparent object for accurate 6DoF pose estimation, it is straight-forward to reconstruct the depth of transparent object first, and then predict the 6DoF pose using ordinary RGB-D pose estimation methods. For the depth reconstruction, we compared two options: (a1) FCRN denotes Fully Convolutional Residual Networks for depth reconstruction [64]. It directly estimates depth using 2D CNN-based network from RGB-channel. (a2) CG denotes ClearGrasp [12], an accurate depth reconstruction algorithm. It estimates the surface normal from the RGB-channel, and then reconstructs the depth by an optimization scheme. For the RGB-D pose estimation methods, we considered two state-of-the-art methods: (b1) DF denotes DenseFusion [2]. It densely fuses the color and geometry features for accurate pose estimation. (b2) RA denotes Rotation Anchor [3]. It is one of the most stable and accurate pose estimation methods for both symmetric and asymmetric objects. By combining the options mentioned above, we had four state-of-the-art baselines: (1) FCRN [64] + RA [3]; (2) FCRN [64] + DF [2]; (3) CG [12] + DF [2]; and (4) CG [12] + RA [3]. In the following, Ours denotes the proposed approach.

The accuracy–threshold curves of the compared methods are shown in Figure 8. The x-axis denotes the varying threshold in terms of ADD-S, and the y-axis denotes the accuracy according to the threshold. The AUC is the area under the accuracy–threshold curve within the range from 0 to 0.1 m. The accuracy of the compared methods are shown in Table 1, in which the threshold is set to 10% of the object diameter. We observed that FCRN-based methods can efficiently reconstruct depth, but the result is not accurate, as the depth is inferred from RGB-channel only. CG-based methods can accurately reconstruct depth, because both RGB-channel and D-channel are considered in the depth
reconstruction process. DF-based methods are stable, even when the estimated depth is not accurate; FCRN + DF still yields sensible result. RA-based methods are accurate; CG + RA outperforms CG + DF when the reconstructed depth is good, but the performance of FCRN + RA degrades when the depth reconstruction is inaccurate. Ours does not rely on depth reconstruction as it directly estimate object pose from estimated normal map, and experimental results show that it is the most accurate among the compared methods.

Figure 8. 3D model of each object and the accuracy–threshold curves of the objects.
Table 1. Pose estimation accuracy of the compared methods. The threshold of ADD-S is set to 10% of the object diameter. Each row corresponds to a type of object. The last row is the evaluation results of all objects. The bold font indicates the best scores.

| Object     | FCRN + RA | FCRN + DF | CG + DF | CG + RA | Ours  |
|------------|-----------|-----------|---------|---------|-------|
| cup        | 12.2      | 40.3      | 76.5    | 71.2    | **88.6** |
| flower     | 13.5      | 53.8      | 76.3    | 80.3    | **92.2** |
| heart      | 7.7       | 35.2      | 28.5    | 73.3    | **88.7** |
| square     | 25.5      | 35.6      | 71.8    | 54.4    | **77.0** |
| stemless   | 23.2      | 32.4      | 69.3    | 62.7    | **80.1** |
| all        | 16.4      | 39.1      | 64.0    | 68.4    | **85.4** |

4.4. Efficiency

The time efficiency of the compared methods are shown in Table 2. We observed that the FCRN-based methods are efficient but not accurate, because the depth is reconstructed from RGB-channel only. The CG-based methods take much longer than FCRN-based methods, since accurate depth is constructed by CG through a time consuming global optimization scheme. The RA-based methods are slightly less efficient than the DF-based ones, as the network structure of RA is more complex than DF. Among the compared methods, Ours is the most efficient, because the depth is not reconstructed and the estimation is directly conducted based on extended point-cloud. The time efficiency of Ours is 0.069 s per instance, and 0.223 s per image (a single image may contain multiple object instances).

Table 2. Time efficiency of the compared methods.

|                      | FCRN + RA | FCRN + DF | CG + DF | CG + RA | Ours  |
|----------------------|-----------|-----------|---------|---------|-------|
| per instance         | 0.108 s   | 0.074 s   | 0.819 s | 0.855 s | 0.069 s |
| per image            | 0.345 s   | 0.234 s   | 2.606 s | 2.715 s | 0.223 s |

4.5. Ablation Study

The extended point-cloud contains three components: the UV code, the normal, and the plane. To study the importance of these components, an ablation study was conducted by evaluating three variations of the extended point-cloud: (1) w/o UV code, which denotes the extended point-cloud without the UV code component; (2) w/o normal, which denotes the extended point-cloud without the normal component; and (3) w/o plane, which denotes the extended point-cloud without the plane component.

The 6DoF pose estimation accuracy of the three variations are shown in Table 3, and the accuracy–threshold curves of the three variations are shown in Figure 9. The observations are as follows: (1) UV encoding is crucial for extended point-cloud. When the UV code is absent, the accuracy of Ours drops from 85.4% to 33.8%. The pose estimator cannot properly work without UV code. (2) Normal estimation is very important for accurate 6DoF transparent object pose estimation. When the normal is absent, the accuracy of Ours drops from 85.4% to 68.8%. (3) Plane estimation is helpful to improve the accuracy. When the plane is absent, the accuracy of Ours drops from 85.4% to 71.5%.

It is also observed that the normal component is closely related to the 3DoF rotation estimation accuracy. To measure the 3DoF rotation accuracy, we replace the estimated translation with ground-truth, and then measure the ADD-S accuracy. The 3DoF rotation accuracy of Ours is 95.2%. It drops to 83.9% when normal is absent, while it slightly drops to 92.0% when plane is absent. It shows that the normal component is more closely related to the 3DoF rotation estimation.
4.6. Qualitative Evaluation

The 6DoF pose estimation results of the compared methods are visualized in Figure 10. Each row corresponds to a scene, and each column corresponds to a method. The first column is the results of CG + DF, the second column is the results of CG + RA, and the third column is the results of Ours. The object pose is shown as tight oriented bounding box of object’s model. Different objects are visualized by boxes with different colors. The results show that Ours accurately estimates transparent object pose in simple scenes (Rows 1 and 6), cluttered scenes (Rows 2 and 4), with occlusions (Rows 4 and 5), and small scale size (Row 6). We also predict truncated objects (partially visible) more accurately than other methods, e.g., the object in the Row 7. Our approach is trained using synthetic images, but the trained model can be robustly applied to real images which are not seen in the training data (please refer to the first four rows of Figure 10).

When several transparent objects occlude each other, the pose estimation results may be unstable in some extent. The failure cases are shown in Figure 11. In Figure 11a, when one transparent object is occluded by another transparent object, the correctness of pose estimation may be affected. In Figure 11b, when multiple small transparent objects overlap each other, several small objects may be falsely estimated as one. As the light paths within mutually occluding transparent objects are complicated, it is very hard to recover the geometry of transparent object behind the transparent object. We will consider to research this complex issue in the future.
Figure 10. The transparent object pose estimation results of CG+DF, CG+RA, and Ours. The object poses are visualized as tight oriented bounding boxes. The first four rows are real images, and the last three rows are synthetic images.
5. Conclusions

In this paper, we present a two-stage approach for 6DoF pose estimation of transparent object from a single RGB-D image. The missing or distorted geometric information caused by depth error is recovered in the first stage, and then the 6DoF pose of transparent object is estimated in the second stage. Specially, an extended point-cloud representation is proposed to estimate object pose accurately and efficiently without explicit depth reconstruction.

In this study, the device we use is the Intel RealSense D435 camera, the same as that of ClearGrasp. The proposed approach is trained and evaluated on the ClearGrasp dataset. It is an interesting topic to discuss whether the proposed approach can be applied to different RGB-D devices. We believe that the proposed approach will work when the following assumptions are satisfied: (1) the RGB-channel and the D-channel are pixel-wisely aligned; (2) the transparent object stands on an opaque plane; and (3) the depth of the opaque plane can be reliably captured by the depth sensor.

If the first assumption is satisfied, no matter what type of depth sensor is used, the noisy depth of the transparent object can be effectively removed, since the segmentation is predicted using only RGB-channel. If the second and third assumptions are satisfied, the plane can be effectively estimated for extended point-cloud extraction. Fortunately, most of the consumer level RGB-D devices satisfy the first and third assumptions. The second assumption holds in many common application scenarios in our daily life. If the above three assumptions are not met, the proposed approach will degrade to the baseline “w/o plane” addressed in Section 4.6. The pose estimation accuracy of “w/o plane” is 71.5. “w/o plane” is less accurate than “Ours”, but it still performs quite well comparing to other state-of-the-art baselines.

Furthermore, we have some notes about the compared baselines: FCRN [64] is a general-purpose depth estimator which takes only RGB-channel as input. CG [12] is an impressive transparent depth recovering method which generates high accurate depth map for robotic grasping, and it can be generalized to unseen objects. DF [2] and RA [3] are among the state-of-the-art ordinary object pose estimators. Surprisingly, the experimental results show that the combinations of these advanced methods do not work very well for 6DoF transparent object pose estimation. The observation suggests that accurate transparent depth estimation with accurate ordinary object pose estimation does not necessarily result in accurate transparent object pose estimation. In this paper, we focus on the specific challenge in the 6DoF transparent object pose estimation problem, and the proposed approach achieves significantly more efficient and accurate performance than the state-of-the-art ones by utilizing the novel extended point-cloud representation. Different from the grasping task [12] which CG aims at, we focus on the 6DoF pose estimation task, which normally assumes that 3D models of objects are known [2,3]; therefore, unlike CG, the proposed approach cannot be applied to unseen objects.

The limitation of the work is that the pose estimation accuracy may degrade when multiple transparent objects occlude each other (i.e., transparent object occludes transparent object), which is a very challenging scenario. In the future, we will apply this technology in robotic teaching and manipulation applications and study the interaction between human hand and the transparent objects.
Author Contributions: Conceptualization, C.X.; methodology, J.C., M.Y., and C.X.; software, J.C. and M.Y.; validation, J.C. and C.X.; formal analysis, C.X.; investigation, C.X. and J.C.; resources, C.X. and L.Z.; writing—original draft preparation, J.C., C.X., and J.Z.; writing—review and editing, C.X. and J.Z.; visualization, J.C.; supervision, C.X., L.Z., and Y.L.; project administration, C.X.; and funding acquisition, C.X. and Y.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China under Grant No. 61876170; the National Natural Science Fund Youth Science Fund of China under Grant No. 51805168; the R&D project of CRRC Zhuzhou Locomotive Co., Ltd. under No. 2018GY121; and the Fundamental Research Funds for Central Universities, China University of Geosciences, No. CUG170692.

Acknowledgments: We thank the volunteers who provided some suggestions for our work. They are Yunkai Jiang and Ming Chen.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Li, S.; Chi, X.; Ming, X. A Robust O(n) Solution to the Perspective-n-Point Problem. *IEEE Trans. Pattern Anal. Mach. Intell.* 2012, 34, 1444–1450. [CrossRef] [PubMed]
2. Wang, C.; Xu, D.; Zhu, Y.; Martin-Martín, R.; Lu, C.; Fei-Fei, L.; Savarese, S. DenseFusion: 6D Object Pose Estimation by Iterative Dense Fusion. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–18 June 2019; pp. 3343–3352. [CrossRef]
3. Tian, M.; Pan, L.; Ang Jr, M.H.; Lee, G.H. Robust 6D Object Pose Estimation by Learning RGB-D Features. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May–4 June 2020. [CrossRef]
4. Zhu, M.; Derpanis, K.G.; Yang, Y.; Brahmbhatt, S.; Zhang, M.; Phillips, C.; Lecce, M.; Daniilidis, K. Single image 3D object detection and pose estimation for grasping. In Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA), Miami, Florida, USA, 20–21 January 2014; pp. 3936–3943. [CrossRef]
5. Tremblay, J.; To, T.; Sundaralingam, B.; Xiang, Y.; Fox, D.; Birchfield, S. Deep object pose estimation for semantic robotic grasping of household objects. *arXiv* 2018, arXiv:1809.10790.
6. Geiger, A.; Lenz, P.; Urtasun, R. Are we ready for autonomous driving? The KITTI vision benchmark suite. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; pp. 3354–3361. [CrossRef]
7. Chen, X.; Ma, H.; Wan, J.; Li, B.; Xia, T. Multi-view 3D Object Detection Network for Autonomous Driving. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 22–25 July 2017; pp. 1907–1915. [CrossRef]
8. Yu, Y.K.; Wong, K.H.; Chang, M.M.Y. Pose Estimation for Augmented Reality Applications Using Genetic Algorithm. *IEEE Trans. Syst. Man Cybern. Part B Cybern.* 2005, 35, 1295–1301. [CrossRef] [PubMed]
9. Marchand, E.; Uchiyama, H.; Spindler, F. Pose Estimation for Augmented Reality: A Hands-On Survey. *IEEE Trans. Vis. Comput. Graph.* 2016, 22, 2633–2651. [CrossRef] [PubMed]
10. 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 Proceedings of the 2016 European Conference on Computer Vision (ECCV), Amsterdam, The Netherlands, 8–16 October 2016. [CrossRef]
11. Li, C.; Bai, J.; Hager, G.D. A Unified Framework for Multi-View Multi-Class Object Pose Estimation. In Proceedings of the 2018 European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 263–281. [CrossRef]
12. Sajjan, S.; Moore, M.; Pan, M.; Nagaraja, G.; Lee, J.; Zeng, A.; Song, S. Clear Grasp: 3D Shape Estimation of Transparent Objects for Manipulation. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May–4 June 2020. [CrossRef]
13. Peng, S.; Liu, Y.; Huang, Q.; Zhou, X.; Bao, H. PVNet: Pixel-Wise Voting Network for 6DoF Pose Estimation. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–18 June 2019; pp. 4561–4570. [CrossRef]
14. Drost, B.; Ulrich, M.; Navab, N.; Ilic, S. Model globally, match locally: Efficient and robust 3D object recognition. In Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA, 13–18 June 2010; pp. 998–1005. [CrossRef]
15. Vidal, J.; Lin, C.; Martí, R. 6D pose estimation using an improved method based on point pair features. In Proceedings of the 2018 4th International Conference on Control, Automation and Robotics (ICCAR), Auckland, New Zealand, 20–23 April 2018; pp. 405–409. [CrossRef]

16. Hinterstoisser, S.; Holzer, S.; Cagniart, C.; Ilic, S.; Konolige, K.; Navab, N.; Lepetit, V. Multimodal templates for real-time detection of texture-less objects in heavily cluttered scenes. In Proceedings of the 2011 International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; pp. 858–865. [CrossRef]

17. Guo, Y.; Bennamoun, M.; Sohel, F.; Lu, M.; Wan, J.; Kwok, N.M. A Comprehensive Performance Evaluation of 3D Local Feature Descriptors. Int. J. Comput. Vis. 2015, 116, 66–89. [CrossRef]

18. Song, S.; Xiao, J. Deep sliding shapes for amodal 3d object detection in rgb-d images. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 808–816. [CrossRef]

19. Park, K.; Mousavian, A.; Xiang, Y.; Fox, D. LatentFusion: End-to-End Differentiable Reconstruction and Rendering for Unseen Object Pose Estimation. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 14–19 June 2020. [CrossRef]

20. Wada, K.; Sucar, E.; James, S.; Lenton, D.; Davison, A.J. MoreFusion: Multi-object Reasoning for 6D Pose Estimation from Volumetric Fusion. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 14–19 June 2020. [CrossRef]

21. Qi, C.R.; Su, H.; Mo, K.; Guibas, L.J. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 22–25 July 2017. [CrossRef]

22. Qi, C.R.; Yi, L.; Su, H.; Guibas, L.J. PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space. In Proceedings of the 31st International Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 5099–5108. [CrossRef]

23. Fritz, M.; Bradski, G.; Karayev, S.; Darrell, T.; Black, M.J. An additive latent feature model for transparent object recognition. Adv. Neural Inf. Process. Syst. 2009, 22, 558–566.

24. Mchenry, K.; Ponce, J.; Forsyth, D. Finding glass. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Diego, CA, USA, 20–26 June 2005; pp. 196–199.

25. Phillips, C.J.; Derpanis, K.G.; Daniilidis, K. A novel stereoscopic cue for figure-ground segregation of semi-transparent objects. In Proceedings of the 2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops), Barcelona, Spain, 6–13 November 2011. [CrossRef]

26. Xie, E.; Wang, W.; Wang, W.; Ding, M.; Shen, C.; Luo, P. Segmenting Transparent Objects in the Wild. arXiv 2020, arXiv:2003.13948.

27. Mchenry, K.; Ponce, J. A Geodesic Active Contour Framework for Finding Glass. In Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), New York, NY, USA, 17–22 June 2006; pp. 1038–1044. [CrossRef]

28. Wang, T.; He, X.; Barnes, N. Glass object localization by joint inference of boundary and depth. In Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012), Tsukuba, Japan, 11–15 November 2012; pp. 3783–3786.

29. Khang, M.P.; Masayuki, M. Transparent object detection using convolutional neural network. In Proceedings of the International Conference on Big Data Analysis and Deep Learning Applications, Miyazaki, Japan, 14–15 May 2018; Springer: Singapore; pp. 86–93. [CrossRef]

30. Lai, P.J.; Fuh, C.S. Transparent object detection using regions with convolutional neural network. In Proceedings of the IPPR Conference on Computer Vision, Graphics, and Image Processing, Taiwan, China, 17–19 August 2015; pp. 1–8.

31. Seib, V.; Barthen, A.; Marohn, P.; Paulus, D. Friend or foe: Exploiting sensor failures for transparent object localization and classification. In Proceedings of the 2016 International Conference on Robotics and Machine Vision; Bernstein, A.V., Olaru, A., Zhou, J., Eds.; Moscow, Russia, 14–16 September 2016; Volume 10253, pp. 94–98. [CrossRef]

32. Han, K.; Wong, K.Y.K.; Liu, M. A Fixed Viewpoint Approach for Dense Reconstruction of Transparent Objects. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015. [CrossRef]
33. Qian, Y.; Gong, M.; Yang, Y. 3D Reconstruction of Transparent Objects with Position-Normal Consistency. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 4369–4377. [CrossRef]

34. Song, S.; Shim, H. Depth Reconstruction of Translucent Objects from a Single Time-of-Flight Camera Using Deep Residual Networks. In Computer Vision–ACCV 2018; Jawahar, C., Li, H., Mori, G., Schindler, K., Eds.; Springer International Publishing: Cham, Switzerland, 2019; pp. 641–657. [CrossRef]

35. Klank, U.; Carton, D.; Beetz, M. Transparent object detection and reconstruction on a mobile platform. In Proceedings of the 2011 IEEE International Conference on Robotics and Automation, Shanghai, China, 9–13 May 2011; pp. 5971–5978. [CrossRef]

36. Eren, G.; Aubreton, O.; Meriaudeau, F.; Secades, L.S.; Fofi, D.; Naskali, A.T.; Truchetet, F.; Ercil, A. Scanning from heating: 3D shape estimation of transparent objects from local surface heating. Opt. Express 2009, 17, 11457–11468. [CrossRef] [PubMed]

37. Ji, Y.; Xia, Q.; Zhang, Z. Fusing depth and silhouette for scanning transparent object with RGB-D sensor. Int. J. Opt. 2017, 2017, 9796127. [CrossRef]

38. Li, Z.; Yeh, Y.Y.; Chandraker, M. Through the Looking Glass: Neural 3D Reconstruction of Transparent Shapes. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 1262–1271. [CrossRef]

39. Albrecht, S.; Marsland, S. Seeing the unseen: Simple reconstruction of transparent objects from point cloud data. In Proceedings of the Robotics: Science and Systems, Berlin, Germany, 24–28 June 2013.

40. Lysenkov, I.; Erhumov, V.; Bradski, G. Recognition and pose estimation of rigid transparent objects with a kinect sensor. Robotics 2013, 273, 273–280. [CrossRef]

41. Lysenkov, I.; Rabaud, V. Pose estimation of rigid transparent objects in transparent clutter. In Proceedings of the 2013 IEEE International Conference on Robotics and Automation, Karlsruhe, Germany, 6–10 May 2013; pp. 162–169. [CrossRef]

42. Guo-Hua, C.; Jun-Yi, W.; Ai-Jun, Z. Transparent object detection and location based on RGB-D camera. J. Phys. Conf. Ser. 2019, 1183, 012011. [CrossRef]

43. Byambaa, M.; Koutaki, G.; Choimaa, L. 6D Pose Estimation of Transparent Object from Single RGB Image. In Proceedings of the Conference of Open Innovations Association, FRUCT, Helsinki, Finland, 5–8 November 2019; pp. 444–447. [CrossRef]

44. Phillips, C.J.; Lecce, M.; Daniilidis, K. Seeing Glassware: From Edge Detection to Pose Estimation and Shape Recovery. In Proceedings of the Robotics: Science and Systems, Ann Arbor, MI, USA, 18–22 June 2016; Volume 3. [CrossRef]

45. Liu, X.; Jonschkowski, R.; Angelova, A.; Konolige, K. KeyPose: Multi-View 3D Labeling and Keypoint Estimation for Transparent Objects. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 11602–11610. [CrossRef]

46. Lysenkov, I.; Erhumov, V. Pose Refinement of Transparent Rigid Objects with a Stereo Camera. In Transactions on Computational Science XIX; Gavrilova, M.L., Tan, C.J.K., Konushin, A., Eds.; Springer: Berlin/Heidelberg, Germany, 2013; pp. 143–157. [CrossRef]

47. Zhou, Z.; Pan, T.; Wu, S.; Chang, H.; Jenkins, O.C. GlassLoc: Plenoptic Grasp Pose Detection in Transparent Clutter. In Proceedings of the 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Macau, China, 3–8 November 2019. [CrossRef]

48. Mathai, A.; Guo, N.; Liu, D.; Wang, X. 3D Transparent Object Detection and Reconstruction Based on Passive Mode Single-Pixel Imaging. Sensors 2020, 20, 4211. [CrossRef] [PubMed]

49. Grammatikopoulou, M.; Yang, G. Three-Dimensional Pose Estimation of Optically Transparent Microrobots. IEEE Robot. Autom. Lett. 2020, 5, 72–79. [CrossRef]

50. Kaiming, H.; Georgia, G.; Piotr, D.; Ross, G. Mask R-CNN. In Proceedings of the IEEE International Conference on Computer Vision, Honolulu, HI, USA, 21–26 July 2017; pp. 2961–2969. [CrossRef]

51. Badrinarayanan, V.; Kendall, A.; Cipolla, R. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. IEEE Trans. Pattern Anal. Mach. Intell. 2019, 39, 2481–2495. [CrossRef] [PubMed]

52. Lin, G.; Milan, A.; Shen, C.; Reid, I. RefineNet: Multi-Path Refinement Networks for High-Resolution Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 22–25 July 2017. [CrossRef]
53. Chen, L.C.; Zhu, Y.; Papandreou, G.; Schroff, F.; Adam, H. Encoder-Decoder with Atrous Separable
Convolution for Semantic Image Segmentation. In Proceedings of the European Conference on Computer
Vision (ECCV), Munich, Germany, 8–14 September 2018. [CrossRef]

54. Schnabel, R.; Wahl, R.; Klein, R. Efficient RANSAC for point-cloud shape detection. In Computer Graphics
Forum; Blackwell Publishing Ltd.: Oxford, UK, 2007; Volume 26, pp. 214–226.

55. Wang, Y.; Sun, Y.; Liu, Z.; Sarma, S.E.; Bronstein, M.M.; Solomon, J.M. Dynamic Graph CNN for Learning on
Point Clouds. ACM Trans. Graph. 2018, 38, 1–12. [CrossRef]

56. Xiang, Y.; Schmidt, T.; Narayanan, V.; Fox, D. PoseCNN: A Convolutional Neural Network for 6D Object
Pose Estimation in Cluttered Scenes. arXiv 2017, arXiv:1711.00199.

57. 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. In Proceedings of the 2016 IEEE Conference on
Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 3364–3372.
[CrossRef]

58. 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 Proceedings of the IEEE International Conference on
Computer Vision, Venice, Italy, 22–29 October 2017; pp. 3848–3856. [CrossRef]

59. Tekin, B.; Sinha, S.N.; Fua, P. Real-Time Seamless Single Shot 6D Object Pose Prediction. In Proceedings
of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017.
[CrossRef]

60. Kehl, W.; Manhardt, F.; Tombari, F.; Ilic, S.; Navab, N. SSD-6D: Making RGB-Based 3D Detection and 6D
Pose Estimation Great Again. In Proceedings of the 2017 IEEE International Conference on Computer Vision
(ICCV), Venice, Italy, 22–29 October 2017; pp. 1521–1529. [CrossRef]

61. Paszke, A.; Gross, S.; Chintala, S.; Chanan, G.; Yang, E.; DeVito, Z.; Lin, Z.; Desmaison, A.; Antiga, L.; Lerer, A.
Automatic differentiation in pytorch. In Proceedings of the 2017 Neural Information Processing Systems
Workshop, Long Beach, CA, USA, 4–9 December 2017.

62. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. arXiv 2014, arXiv:1412.6980.

63. Hinterstoisser, S.; Lepetit, V.; Ilic, S.; Holzer, S.; Navab, N. Model Based Training, Detection and Pose
Estimation of Texture-Less 3D Objects in Heavily Cluttered Scenes. In Proceedings of the Asian Conference
on Computer Vision. Daejeon, Korea, 5–9 November 2012; pp. 548–562. [CrossRef]

64. Laina, I.; Rupprecht, C.; Belagiannis, V.; Tombari, F.; Navab, N. Deeper depth prediction with fully
convolutional residual networks. In Proceedings of the 2016 Fourth international conference on 3D vision
(3DV), Stanford, CA, USA, 25–28 October 2016; pp. 239–248. [CrossRef]

Publisher’s Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional
affiliations.

© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution
(CC BY) license (http://creativecommons.org/licenses/by/4.0/).