6D Pose Estimation of Transparent Object from Single RGB Image

Munkhtulga Byambaa, Gou Koutaki
Kumamoto University
Kumamoto, Japan
munkhtulga, koutaki@navi.cs.kumamoto-u.ac.jp

Lodoiravsal Choimaa
National University of Mongolia
Ulaanbaatar, Mongolia
lodoiravsal@num.edu.mn

***Abstract*—Transparent objects are one of the most common objects in everyday life. Estimating pose of these objects are required to pick and manipulate such objects. However, due to the absorption and refraction of light, it is hard to capture depth image of transparent object. In this paper, we address this problem using synthetic dataset to train deep neural network and estimate pose of known transparent objects. Synthetic dataset contains depth map of transparent object which we created in realistic looking environment. Also combining domain randomized and photorealistic images, we create desired amount of annotated data in order to network operate successfully against real world data. We conducted experiment on 3D printed transparent objects in the real environment. For future work, we are planning to build random bin picking system for transparent object.**

I. Introduction

In era when automation is developing rapidly, many fields are using robots for hard and repetitive tasks starting from service robots in home environment to industrial robots in assembly line. Especially bin and shelve picking is getting attention for its possible applications. Recognition and pose estimation is needed for pick-and-place robot. In recent years, there have been many researches and datasets about 6DOF pose estimation. Some methods use traditional feature point matching between 3D models and images [1], [2], [3]. Other methods which use deep neural networks have been applied too [4], [5], [6]. But many of them did not consider transparent object much.

Transparent objects are one of the most common objects in everyday life at home or industry. However, recognizing and estimating pose of transparent objects are still challenging task in robot vision even after emergence of 3D sensors. Transparent objects are difficult to detect due to the appearance of transparent objects that can vary in different backgrounds. We can use the modern 3D sensors (Kinect, Asus Xtion or RealSense), which provides RGB and depth information simultaneously, but it cannot capture reliable depth data on the transparent object surfaces. In order to solve this problem, we present the recognition and pose estimation of transparent objects based on synthetic data in this paper.

Synthetic data can very useful for addressing certain problems in 6DOF pose estimation. In 3D object detection and pose estimation, dataset is important and it is difficult to label real images manually. Synthetic data can be used in deep neural network training due to the generation of unlimited amount of annotated data. Different from 2D object detection which bounding boxes are relatively easy to annotate, 3D object detection requires labeled data that is near impossible to generate manually. Even though it is possible to semi-automatically label data (using a tool such as LabelFusion [7]), the time-consuming nature of the task makes it difficult to generate training data with sufficient variation. To overcome these limitations of real data, we turn to synthetically generated data. Specifically, we use a combination of domain randomized (DR) data and photorealistic data to take advantage of the strengths of both.

We have used 3D printed transparent object in our work and reason is its fully customizable in scope of shape and size. In summary, our work has the following contributions:

- We propose method for transparent object 6D pose estimation using synthetic data to complete unreliable depth image captured by 3D sensors. To our knowledge, this is the first time that synthetic data is being used for transparent object pose estimation.
- Synthetic data also can be easy to annotate since there are not datasets dedicated to transparent objects and when we are using custom objects.

This paper is as follows. In next chapter, we will discuss the previous works. Then, we propose method to synthetic dataset, followed by experimental results and conclusion.

II. Related works

A. Transparent objects recognition and pose estimation

An appearance of transparent objects can vary in different backgrounds. Transparent objects usually don't have their own texture features, their edges are normally weak and intensity gradient features are heavily influenced by seeing through background clutter. Meaning that classical 2D computer vision algorithms for recognition and pose estimation are difficult to apply to transparent objects. An Additive Latent Feature Model is used for Transparent Object Recognition as [8] showed, they used a LDA-SIFT model, [9] proposed a new feature called the light field distortion (LFD) feature. Based on [9], [10] showed an LFD feature and use it to segment the transparent objects from RGB image, and it got a better result. [10] employed a probabilistic formulation for segmentations glass regions.

3D point clouds have been used successfully for object recognition and pose estimation. However, modern 3D sensors (Structured light, ToF, stereo cameras or laser scanners) can't estimate depth reliably and produce point clouds for

transparent and specular objects so these algorithms cannot be applied. To solve these problems, [11] have divided problems into segmentation, pose estimation and recognition tasks. Then, unknown transparent objects are segmented from a single image of Kinect sensor by exploiting its failures on specular surfaces. Next, 3D models of objects created at the training stage are fitted to extracted edges. Finally, a cost function value is used to make a decision about an instance of the object and determine its 6DOF pose relative to the robot.

In [12], they used RGB-D image and IR image to localize and detect transparent object. In detection task, RealSense is employed to retrieve the transparent candidates from the depth image and the corresponding candidates in the RGB image and IR image are then extracted separately. Then, they used SIFT features to recognize the transparent ones from the candidates. In location process, they obtained a new group of RGB images and IR images by adjusting camera orientation to make its optical axis perpendicular to the normal direction of the plane on which the object is placed. The object contours in RGB image and IR image are then extracted, respectively. The three-dimensional object is finally reconstructed by means of stereo matching of the two contours, and the current pose information of the object is calculated in the end.

Cross-modal stereo can be used to get depth estimation on transparent objects with Kinect as in [15], but its quality is lessor. [15] showed the method of using missing depth information to segment the object and pose estimate which first proposed by [18], [17] used Geometric Hashing to solve the clutter. [13] reconstructed objects by moving around a transparent object in the scene, [14] combined multiple sensor modalities, [16] showed a method of using seashell sensors to add missing point cloud data. [18] improved the Cross-modal stereo by using fully-connected CRF.

B. Synthetic data for training

Given the huge need for massive amounts of annotated training data, a recent research trend has shifted to providing synthetic datasets for training [19], [20], [21]. Most of these datasets are photorealistic, thus requiring significant 3D modeling skill. To solve this challenge, domain randomization [22], [23] has been proposed as a reasonable alternative that forces the network to learn to focus on essential features of the data by randomizing the training input in non-realistic ways. While domain randomization has shown promising results on several tasks, it is yet to produce state-of-the-art results compared with real-world data. In [22], [24], for example, the authors concluded that fine-tuning with real data was necessary for domain randomization to compete with real data.

[6] proved that domain randomization alone is not sufficient for the network to fully understand a scene, given its non-realism and lack of context. Thus, their approach of using photorealistic data to complement domain randomization was effective approach to this problem.

III. Proposed method

We used synthetic dataset for pose estimation of transparent 3D printed objects. Since we cannot capture depth map reliably with 3D sensors, we created transparent 3D model. Similar to [1], [2], [3], we used CAD model to train network. But even in game engine, depth map cannot be captured since depth camera will capture value of pixel behind transparent material. To solve that, we duplicated 3D model with opaque material which is not shown in RGB camera but shows in depth camera. Synthetic depth image is shown in Fig. 1.

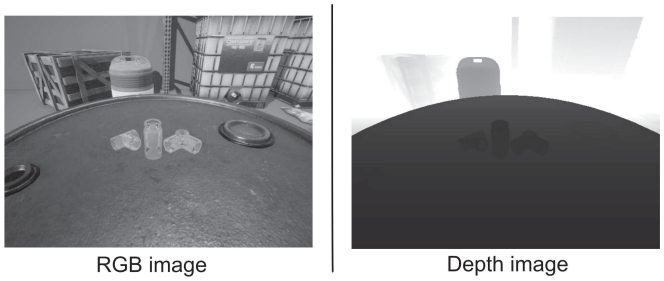

Fig. 1. Synthetic depth captured in game engine

A. Network architecture

In this work, we adopted method in [6]. Their one-shot fully convolutional deep neural network detects keypoints using a multistage architecture. The feedforward network takes as input an RGB image of size w x h x 3 and branches to produce two different outputs, belief maps and vector fields. There are nine belief maps, one for each of the projected 8 vertices of the 3D bounding boxes, and one for the centroids. Similarly, there are eight vector fields indicating the direction from each of the 8 vertices to the corresponding centroid, to enable the detection of multiple instances of the same type of object. (In this case, w = 640, h = 480).

B. Detection and pose estimation

After the network has processed an image, it is necessary to extract the individual objects from the belief maps. This approach [6] relies on a simple postprocessing step that searches for local peaks in the belief maps above a threshold, followed by a greedy assignment algorithm that associates projected vertices to detected centroids. For each vertex, this latter step compares the vector field evaluated at the vertex with the direction from the vertex to each centroid, assigning the vertex to the closest centroid within some angular threshold of the vector. Once the vertices of each object instance have been determined, a PnP algorithm is used to retrieve the pose of the object. This step uses the detected projected vertices of the bounding box, the camera intrinsics, and the object dimensions to recover the final translation and rotation of the object with respect to the camera. All detected projected vertices are used, as long as at least the minimum number (four) are detected.

IV. Experimental Results

In this work, we experimented on simulation objects and real 3D printed objects. For simulation experiment, we created test dataset which has 100 images for each object. Results are shown in the Fig. 2 and Table. I. Poses are estimated with the accuracy of several pixels. For pose estimation error, we have calculated average Euclidean distance of all model points between ground truth and estimated pose.

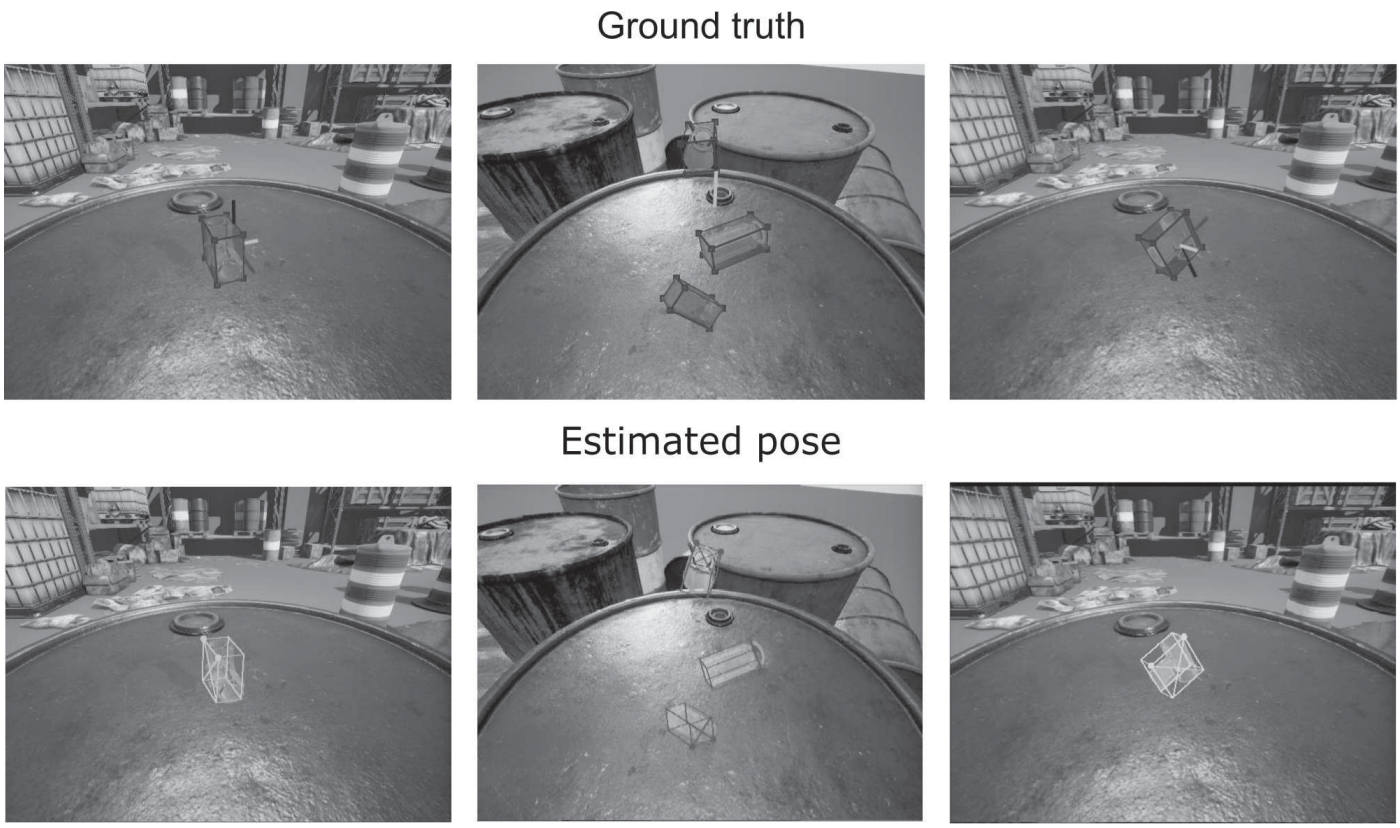

Fig. 2. **Simulation results of pose estimation on custom transparent object**. First column is 3-way tube. Next one is bottle and the last one is T-joint. Top: Ground truth, which was annotated on NDDS [25]. Bottom: Our method trained on 6k photorealistic and domain randomized data.

A. Dataset

Dataset was done on NDDS plugin [25] in Unreal Engine 4 which can capture color and depth image with annotations at rate of 50-100 Hz. Also, class and instance segmentation can be captured. Using this tool, we prepared our custom dataset with 3 transparent objects.

Our custom dataset consists of 2 types as shown in Fig. 3. One is domain randomization which are generated by randomly changing, overlaid textures, backgrounds, object poses, lighting, and noise. Other one is photorealistic images which was captured in real looking environments. As mentioned earlier, we printed custom 3D objects using clear resin on SLA printer. 3 objects are 3-way tube, bottle and T-joint. These objects were selected due to the frequent occurrence in assembly line, graspability and transparency.

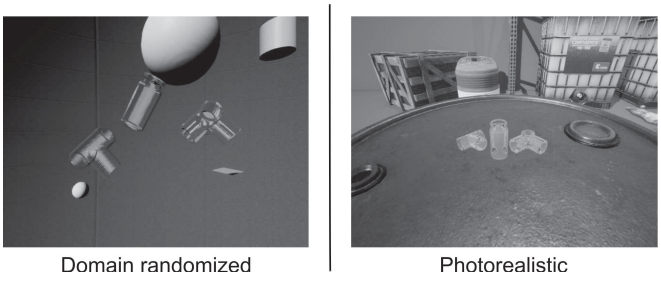

Fig. 3. Synthetic dataset used in training

B. Training

For training, we used 6k photorealistic image (PR) and domain randomized (DR) image which has multiple instances of same object. During the training process, we varied amount of image in dataset. We trained network on DR data only using 2k, 6k, 18k, 30k. (We voluntarily chose the 3-way tube for this experiment.) The biggest performance increase occurred from 2k to 6k and the highest value was achieved with 30k. This experiment was then repeated using photorealistic images instead, highest value was also achieved at 30k. For data augmentation, Gaussian noise (σ = 2:0), random contrast (σ = 0:2) and random brightness (σ = 0:2) were added. The network was implemented using PyTorch v1.0. The VGG-19 feature extractions were taken from publicly available trained weights in torchvision open models. The networks were trained for 60 epochs with a batchsize of 32. Adam was used as the optimizer with learning rate set at 0.001. The system was trained on an NVIDIA Tesla V100 with 16GB memory and testing used an NVIDIA GeForce 1080 Ti 8GB.

TABLE I. AVERAGE ERROR OF THE POSE ESTIMATION (SINGLE OBJECT)

Object	Pose estimation error (pixel)
3-way tube	21.4
Bottle	27.6
T-joint	27.4

For real 3D objects, we have conducted few tests as shown in Fig. 4. However, we haven't calculated average pose estimation error due to the ground truth had a systematic error.

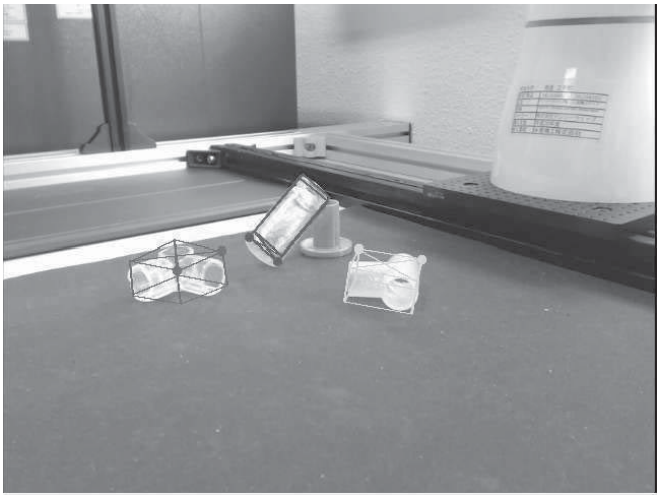

Fig. 4. Pose estimation of custom 3D printed transparent object

V. Conclusion and future works

It is inevitable to skip transparent object when we are trying to grasp common objects in house or industry. In many works, researchers have been using lack of depth image as advantage to locate transparent object. In this work, we have proposed simple method to complement depth image which is needed for effective pose estimation. We used CAD to model 3D object in game engine. But even in game engine, depth map cannot be captured since depth camera will capture value of pixel behind transparent material. To solve that, we duplicated 3D model with opaque material which is not shown in RGB camera but shows in depth camera.

For future work, we are planning to use estimated pose for random bin picking system. Our method is not effective against too reflective or symmetric object. Also, other transparent object pose estimation methods could not handle overlapping transparent objects. These challenges should be main direction for transparent object recognition and pose estimation.

Acknowledgment

This work was supported by ”Higher Engineering Education Development” Project - Research and Development for Power Electronics and Industrial Automation (J14C16).

References

[1] Y. Chen, G. Sun, H. Lin and S. Chen, “Random bin picking with multiview image acquisition and cad-based pose estimation”, *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 2218-2223, Oct 2018.

[2] R. He, J. Rojas and Y. Guan, “A 3D object detection and pose estimation pipeline using RGB-D images”, *2017 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, Macau, 2017, pp. 1527-1532.

[3] C. Wu, S. Jiang and K. Song, “CAD-based pose estimation for random bin-picking of multiple objects using a RGB-D camera”, *2015 15th International Conference on Control, Automation and Systems (ICCAS)*, Busan, 2015, pp. 1645-1649.

[4] Yu Xiang, Tanner Schmidt, Venkatraman Narayanan and Dieter Fox, “PoseCNN: A Convolutional Neural Network for 6D Object Pose Estimation in Cluttered Scenes”, *Robotics: Science and Systems (RSS)*, 2018.

[5] Chen Wang, Danfei Xu, Yuke Zhu, Roberto Martín-Martín, Cewu Lu, Li Fei-Fei and Silvio Savarese, “DenseFusion: 6D Object Pose Estimation by Iterative Dense Fusion”, *Computer Vision and Pattern Recognition (CVPR)*, 2019.

[6] Jonathan Tremblay, Thang To, Balakumar Sundaralingam, Yu Xiang, Dieter Fox and Stan Birchfield, “Deep object pose estimation for semantic robotic grasping of household objects”, *Conference on Robot Learning (CoRL)*, 2018.

[7] Pat Marion, Peter R. Florence, Lucas Manuelli and Russ Tedrake, “A Pipeline for Generating Ground Truth Labels for Real RGBD Data of Cluttered Scenes”, *ICRA*, 2017.

[8] Fritz Mario, Gary Bradski, Sergey Karayev, Darrell Trevor and Michael J. Black, “An Additive Latent Feature Model for Transparent Object Recognition”, *Advances in Neural Information Processing Systems 22*, 2009, pp. 558-566.

[9] K. Maeno, H. Nagahara, A. Shimada and R. Taniguchi, “Light Field Distortion Feature for Transparent Object Classification”, *2013 IEEE Conference on Computer Vision and Pattern Recognition*, Portland, OR, 2013, pp. 2786-2793.

[10] Y. Xu, H. Nagahara, A. Shimada and R. Taniguchi, “TransCut: Transparent Object Segmentation from a Light-Field Image”, *2015 IEEE International Conference on Computer Vision (ICCV)*, Santiago, 2015, pp. 3442-3450.

[11] Nicholas Roy, Paul Newman and Siddhartha Srinivasa, ”Recognition and Pose Estimation of Rigid Transparent Objects with a Kinect Sensor”, *Robotics: Science and Systems VIII*. MITP, 2013.

[12] Guo-Hua Chen, Jun-Yi Wang and Ai-Jun Zhang, ”Transparent object detection and location based on RGB-D camera”, *Journal of Physics: Conference Series*. vol.1183, 2019, pp. 012011.

[13] N. Alt, P. Rives and E. Steinbach, ”Reconstruction of transparent objects in unstructured scenes with a depth camera”, *2013 IEEE International Conference on Image Processing*, Melbourne, VIC, 2013, pp. 4131-4135.

[14] Alexander Hagg, Frederik Hegger and Paul-Gerhard Plöger, “On Recognizing Transparent Objects in Domestic Environments Using Fusion of Multiple Sensor Modalities”, *RoboCup*, 2016.

[15] Walon Wei-Chen Chiu, Ulf Blanke and Mario Fritz, “Improving the Kinect by Cross-Modal Stereo”, *BMVC*, 2011.

[16] K. Huang, L. Jiang, J. R. Smith and H. J. Chizeck, “Sensor-aided teleoperated grasping of transparent objects”, *2015 IEEE International Conference on Robotics and Automation (ICRA)*, Seattle, WA, 2015, pp. 4953-4959.

[17] I. Lysenkov and V. Rabaud, “Pose estimation of rigid transparent objects in transparent clutter”, *2013 IEEE International Conference on Robotics and Automation*, Karlsruhe, 2013, pp. 162-169.

[18] M. Ulrich, C. Wiedemann and C. Steger, “CAD-based recognition of 3D objects in monocular images”, *2009 IEEE International Conference on Robotics and Automation*, Kobe, 2009, pp. 1191-1198.

[19] A. Dosovitskiy, P. Fischer, E. Ilg, P. Häusser, C. Hazirbas, V. Golkov, P. Smagt, D. Cremers, and T. Brox, “FlowNet: Learning optical flow with convolutional networks”, *ICCV*, 2015.

[20] J. McCormac, A. Handa, and S. Leutenegger, “SceneNet RGB-D: 5M photorealistic images of synthetic indoor trajectories with ground truth”, *ICCV*, 2017.

[21] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. Lopez, “The SYNTHIA dataset: A large collection of synthetic images for semantic segmentation of urban scenes”, *CVPR*, 2016.

[22] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, “Domain randomization for transferring deep neural networks from simulation to the real world”, *IROS*, 2017.

[23] F. Sadeghi and S. Levine, “CAD2RL: Real single-image flight without a single real image”, *Robotics: Science and Systems (RSS)*, 2017

[24] J. Tremblay, A. Prakash, D. Acuna, M. Brophy, V. Jampani, C. Anil, T. To, E. Cameracci, S. Boochoon, and S. Birchfield, “Training deep networks with synthetic data: Bridging the reality gap by domain randomization”, *CVPR Workshop on Autonomous Driving (WAD)*, 2018.

[25] Jonathan Tremblay, Thang To and Stan Birchfield, “Falling Things: A Synthetic Dataset for 3D Object Detection and Pose Estimation”, *CVPR Workshop on Real World Challenges and New Benchmarks for Deep Learning in Robotic Vision*, 2018.