KeyPose: Multi-View 3D Labeling and Keypoint Estimation for Transparent Objects

Xingyu Liu[1*] Rico Jonschkowski[2] Anelia Angelova[2] Kurt Konolige[2]
[1]Stanford University [2]Robotics at Google

Abstract

Estimating the 3D pose of desktop objects is crucial for applications such as robotic manipulation. Many existing approaches to this problem require a depth map of the object for both training and prediction, which restricts them to opaque, lambertian objects that produce good returns in an RGBD sensor. In this paper we forgo using a depth sensor in favor of raw stereo input. We address two problems: first, we establish an easy method for capturing and labeling 3D keypoints on desktop objects with an RGB camera; and second, we develop a deep neural network, called KeyPose*, that learns to accurately predict object poses using 3D keypoints, from stereo input, and works even for transparent objects. To evaluate the performance of our method, we create a dataset of 15 clear objects in five classes, with 48K 3D-keypoint labeled images. We train both instance and category models, and show generalization to new textures, poses, and objects. KeyPose surpasses state-of-the-art performance in 3D pose estimation on this dataset by factors of 1.5 to 3.5, even in cases where the competing method is provided with ground-truth depth. Stereo input is essential for this performance as it improves results compared to using monocular input by a factor of 2. We will release a public version of the data capture and labeling pipeline, the transparent object database, and the KeyPose models and evaluation code. Project website:* https://sites.google.com/corp/view/keypose.

1. Introduction

Estimating the position and orientation of 3D objects is one of the core problems in computer vision applications that involve object-level perception such as augmented reality (AR) and robotic manipulation. Rigid objects with a known model can be described by 4D pose (e.g., vehicles [13, 10]), 6D pose [31, 5], and 9D pose where scale is predicted [29]. A more flexible method uses 3D *keypoints* [16, 26], which can handle articulated and deformable objects such as the human hand or body [25, 12]. While some of these methods predict 3D keypoints from a single RGB image, others use RGBD data collected by a depth sensor [28, 16, 3] to achieve better accuracy. Unfortunately, existing commercial depth sensors, such as projected light or time-of-flight (ToF) sensors, assume that objects have opaque, lambertian surfaces that can support diffuse reflection from the sensor. Depth sensing fails when these conditions do not hold, e.g., for *transparent* or *shiny metallic* objects. Figure 1 shows such an example.

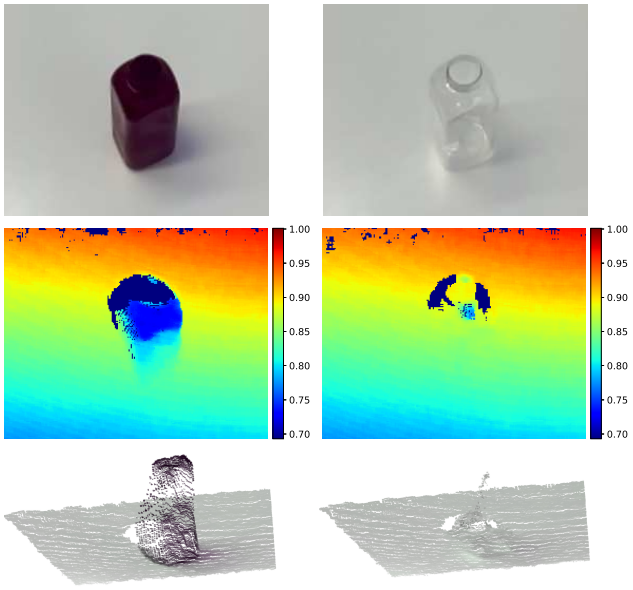

Figure 1: RGB image (top), depth map (middle), and point cloud (bottom) of an opaque bottle (left) and its transparent twin (right). The opaque bottle returns reasonable depth while the transparent one returns invalid depth values using a Microsoft Azure Kinect sensor.

In this paper, we present the first method of **keypoint-based pose estimation for (transparent) 3D objects from stereo RGB images**. There are several challenges: first, there is no available large-scale dataset for transparent 3D

*Work done as an intern at Google Research/Robotics at Google.

object pose estimation from stereo images with annotated keypoints. Datasets such as NYUDepth v2 [19] lack annotations for precise pose of each individual objects, while other datasets such as LabelFusion [16], YCB dataset [3] and REAL275 [29] annotate monocular RGBD images of opaque objects. The second challenge is the annotation of pose of transparent 3D objects. Existing datasets such as [16, 3, 29] require accurate depth information as well as an object CAD model so that alignment algorithms such as iterative closest point (ICP) [2] can be applied. The third challenge is how to leverage only RGB images for 3D keypoint estimation, thus obviating the need for a depth sensor.

To address the challenges regarding data acquisition and annotation, we introduce an efficient method of capturing and labeling stereo RGB images for transparent (and other) objects. Although our method does not need them, we also capture and register depth maps of the object, for both the transparent object and its opaque twin, registered with the stereo images; we use a robotic arm to help automate this process. The registered opaque depth allows us to compare to methods that require depth maps as input such as DenseFusion [28]. Following the proposed data capturing and labeling method, we constructed a large dataset consisting of 48k images from 15 transparent object instances. We call this dataset TOD (Transparent Object Dataset).

To reduce the requirement on reliable depth, we propose a deep model, KeyPose, that predicts 3D keypoints on transparent objects from cropped stereo RGB input. The crops are obtained from a detection stage that we assume can loosely bound objects (see [23] for an appropriate method for transparent objects). The model determines depth implicitly by combining information from the image pair, and predicting the 3D positions of keypoints for object instances and classes. After training on TOD, we compare KeyPose to the best existing RGB and RGBD methods and find that it vastly outperforms them on this dataset. In summary, we make the following contributions:

- A pipeline to label 3D keypoints on real-world objects, including transparent objects that does not require depth images, thus making learning-based 3D estimation of previously unknown objects possible without simulation data or accurate depth images. This pipeline supports a twin-opaque technique to enable comparison with models that require depth input.
- A dataset of 15 transparent objects in 6 classes, labeled with relevant 3D keypoints, and comprising 48k stereo and RGBD images with both transparent and opaque depth. This dataset can also be used in other transparent 3D object applications.
- A deep model, KeyPose, that predicts 3D keypoints on these objects with high accuracy using RGB stereo input only, and even outperforms methods which use ground-truth depth input.

2. Related Work

4D/6D/9D Pose Representation. The assumption behind these representations is the rigidity of the object, so that translation, rotation and size is sufficient to describe its configuration. Existing techniques for 4D/6D/9D pose estimation can generally be categorized by whether a 3D CAD model is used in training or inference. The first type of technique aligns the observed RGB images with rendered CAD model images [5, 11], or aligns the observed 3D point clouds with 3D CAD model point clouds with algorithms such as ICP [28], or renders mixed reality data from 3D CAD models as additional training data [29]. While it is possible to render high-quality RGB scenes of transparent objects using ray-tracing, there has been no work done on rendering depth images that faithfully reproduces the degraded depth seen in real-world RGBD data (see Figure 1).

The second type of technique regresses the object coordinate values from the RGB image or 3D point clouds [31, 13, 10, 21, 22]. Our method does not assume object rigidity, and the object pose is based the locations of 3D keypoints, which can be used on articulated or deformable objects. Our method also does not rely on prior knowledge about each individual object, such as a 3D CAD model.

Keypoint Based Pose Representation. Previous work has explored deep learning methods for detecting 3D keypoints of an object given a monocular RGB image [26] or RGBD image [15]. The core is to predict probability maps for the 2D keypoint locations, and then use the given or predicted depth image for 3D. Other works proposed similar methods for monocular pose estimation [25, 18, 27]. Though estimating 3D positions from a single RGB image is an ill-conditioned problem, these methods implicitly learn the prior of object size during training, or rely on the known object 3D model. Our method is inspired by these works and focuses on 3D keypoint location estimation from stereo instead of single images, and is well-conditioned even for similar objects that vary in scale. Recently, a method similar to ours was proposed for hand tracking using raw strereo [12]. For rigid objects with a known model, the 6D pose can be recovered using the Procrustes algorithm (see the Supplementary materials).

Stereo for Disparity Estimation. Estimating disparity and therefore depth from stereo has been a long-standing problem in computer vision. The success of deep-learning methods in computer vision inspired research in this area, using end-to-end deep networks equipped with a correlation cost volume [17, 9, 6, 32], or point-based depth representation and iterative refinement [4]. Here, instead of generating a dense disparity field, we focus on estimating the 3D location of sparse keypoints directly from stereo images.

3D Object Pose Estimation Datasets. Directly labeling 3D object pose in real RGB images is costly. All existing real (non-synthetic) datasets for 3D object pose estima-

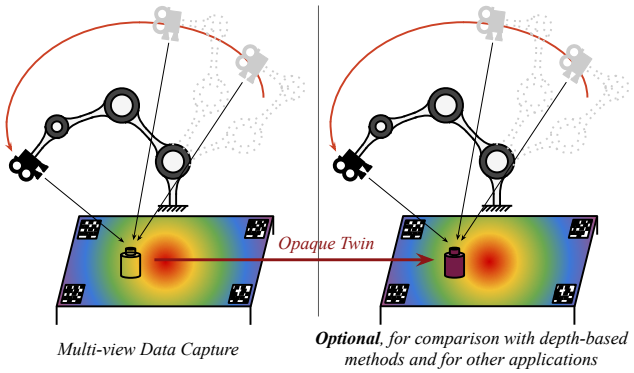

Figure 2: Data capturing pipeline. We mount both the stereo RGB camera and RGBD camera on the end-effector of the robot. We then use the robot arm to perform similar paths to scan both the opaque lambertian object (left) and its transparent twin placed at the same location of a textured surface (right). AprilTags [30] are used as global pose indicator for the cameras.

tion rely on capturing RGBD images and annotating pose by either constructing a 3D mesh [15], or fitting 3D CAD models to 3D point clouds [16, 3, 29, 7], neither of which is possible for transparent objects. On the contrary, we build a data capturing pipeline where ground-truth depth of transparent object keypoints can be efficiently obtained, without relying on depth or 3D CAD models.

Estimation of transparent and reflective objects. Objects that are transparent or reflective present significant challenges for all camera-based depth estimation. Works on estimating transparent object pose and geometry might assume knowing object 3D model [20, 14] or rely on synthetic data to train vision models [24, 23]. Our data capturing and labeling enables generating large-scale real dataset for training and testing transparent object pose and geometry, so synthetic data are not needed.

3. Transparent Object Dataset (TOD)

In this section, we describe the data capturing pipeline that enables efficient capture and labeling of 3D keypoints for a large number of samples without requiring a depth sensor.

3.1. Data Collection with a Robot

Hand-labeling 3D keypoints in individual RGB images is difficult or impossible due to uncertainty about keypoint depth. Instead, we leverage multi-view geometry to raise 2D labels from a small number of images into 3D labels for a set of images where the object has not moved. The general idea is illustrated in Figure 2.

We use a stereo camera with known parameters to capture images in a sequence, moving the camera with a robot arm (we could also move it by hand). To estimate the pose of the camera relative to the world, we set up a planar form with AprilTags [30] that can be be recognized in an image, and from their known locations estimate the camera pose. From a small subset of widely-separated poses, we label 2D keypoints on the object. Optimization from multi-view geometry gives the 3D position of the keypoints, which can be reprojected to all images in the sequence. To increase diversity, we place various textures under the object. Figure 3 shows some challenging data examples.

Figure 3: Challenging cases in our dataset, including dark background textures (left), thin handles of mugs (middle) and motion blur (right). Accurately locating these objects is a difficult task even for human.

The resultant labeled stereo samples are sufficient to train and evaluate the KeyPose model. We can collect and label data for a new object in a few hours. In addition to the stereo data, we also capture and register depth data using the Microsoft Kinect Azure RGBD device. This data is purely ancillary to our model, but it lets us compare KeyPose to methods that require depth data. We collect two depth images, one during the initial scan with co-mounted stereo and RGBD devices, and one with the transparent object replaced by its opaque (painted) twin during a second scan (Figure 2, right). Although the RGBD images are captured at slightly different poses from the stereo (due to variations in the trajectories and camera capture times), we can leverage the calculated pose of the RGBD camera (using AprilTags in the RGB image), and the known offset of the depth sensor from the RGB sensor, to warp the depth image to align precisely with the left stereo image (see Figure 1).

3.2. Keypoint Labeling and Automatic Propagation

To accurately construct this dataset, we need to address different sources of error. First, since AprilTag detection is imperfect in finding tag positions, we spread out these tags on the target to produce large baselines for camera pose estimation. Second, since human labeling of keypoints on 2D images introduces error, we use a farthest-point algorithm on the camera poses to ensure that annotated images used in going from 2D to 3D have a large baseline.

We want to know the accuracy of the manual annotation. While the absolute ground truth of the 3D keypoints is unknown, we can estimate the labeling error, given the known reprojection errors of the AprilTags and 2D annotations. Using a Monte Carlo simulation based on the repro-