3D Pose Estimation for the Object with Knowing Color Symbol by Using Correspondence Grouping Algorithm

Erika Miyake, Tomohito Takubo, Atsushi Ueno

<table>
<thead>
<tr>
<th>Citation</th>
<th>IEEE/SICE International Symposium on System Integration (SII). 2020: 19431607</th>
</tr>
</thead>
<tbody>
<tr>
<td>Date of Conference</td>
<td>2020-01-12^15</td>
</tr>
<tr>
<td>Issue Date</td>
<td>2020-03-09</td>
</tr>
<tr>
<td>Type</td>
<td>Conference paper</td>
</tr>
<tr>
<td>Textversion</td>
<td>author</td>
</tr>
<tr>
<td>Rights</td>
<td>© 2020 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.</td>
</tr>
<tr>
<td>DOI</td>
<td>10.1109/SII46433.2020.9025968</td>
</tr>
</tbody>
</table>

Self-Archiving by Author(s)
Placed on: Osaka City University Repository
3D pose estimation for the object with knowing color symbol by using correspondence grouping algorithm

Erika Miyake¹ and Tomohito Takubo¹ and Atsushi Ueno¹

Abstract—Color feature matching for point cloud data with color 2D code is proposed to recognize the object whose unique feature is difficult to be made by only using shape information. When a specific object such as a commercial product is recognized, it is possible to inquire the product information by attaching a known marker registered in the network server. However, even when a known marker is attached, it is difficult to acquire the accurate marker position and orientation unless the image is taken from a direction that allows good recognition. On the other hand, the position/orientation estimation method using point cloud information cannot be effective unless the shape information is easy to appear. In the proposed method, in order to obtain product information and accurately measure the position and orientation of the product, a marker that is given a characteristic color is attached to a known location on the target object so that the color information as well as shape information can be evaluated simultaneously. In the case of packaged products, it is not necessary to give color information in particular. However, the artificial known marker could be the important evaluation item to estimate object posture since the products that are covered with a transparent film such as a sandwich and a rice ball in the convenience store have different textures for each individual. In order to evaluate the accuracy of the proposed method, the recognition results are compared in a 3D virtual environment. For the evaluation of matching with corresponding grouping algorithm, two major feature-based matching algorithm are implemented; one is 3D-HV (Hough Voting) which returns acceptable inlier searching performance in real-time, and another is RANSAC (Random Sample Consensus) which is known as the high precision performance. In addition, 2.5D and complete 3D reference model are prepared to compare the computational time and the accuracy of the posture estimation. As a result of the experiment, the method using the CSHOT feature, the RANSAC and the complete 3D model as the search method has the highest accuracy and is practical as the search time.

I. INTRODUCTION

Robots are expected to be used in various fields such as daily life, society, industry. To enhance robot technology in a competitive environment, various robot competitions are held in the world. Our laboratory participated in FCSC(Future Convenience Store Challenge) held in WRS (World Robot Summit) 2018, and demonstrated the display and disposal of products using a mobile manipulator. Object manipulation is one of the most important technology, and it required precise object pose estimation to manipulate the object with a robot hand. Object pose estimation is very important because it plays the role of the robot’s eyes when the robot manipulates objects. There are many approaches such as estimation with 2D images [2], and the combination of 2D contour and 3D shape primitives [3]. The approach of deep learning using RGB-D images [4] and multi-view 2D images [5] as inputs are popular topic. Template-based object recognition techniques are mainly used for tasks that have a specified target. Research on the template base 3D object pose estimation using point cloud data captured by using depth camera is a very hot topic. In this paper, the template base object pose estimation is focused since identification and position/posture recognition of specific objects are required in the Future Convenience Store Challenge.

Correspondence Grouping Algorithm[6] is one of the powerful object detection method using the reference model of 3D point cloud data. In this method, feature descriptors play the important role of feature matching. 3D feature descriptors are divided into two groups, local descriptors and global descriptors. Typical local descriptors are Spin Image[7], PPF[8], SHOT[9], CSHOT[14], FPFH[10] that use point cloud data around the keypoints, and PPF[11] that use geometric relation of points, so that they describe the patches of local shape of an object by point group. On the other hand, global descriptors describe the whole of an object features. Global features include shape features and texture features. VFH[12] is example of global descriptor. Using these feature descriptors, the corresponding points between a point on a reference model and a point on a scene are estimated by a matching algorithm. A method with a high inlier rate of corresponding points is assumed as high accuracy of pose estimation.

Many of these 3D pose estimation studies use the target object whose shape is complicated, so that it is easy to find the specific feature points and match between them. However, daily products have simple shapes that consist of only flat surfaces and these objects cause false recognition. For example, in the task of FCSC mentioned above, it was necessary to display and discard sandwiches which have flat surfaces. For the object composed of flat planes, it is presumed that it is difficult to select correct correspondences because similar features are described on each side. Therefore, accurate object pose estimation is required for objects composed of flat planes. For example, in the task of FCSC, the sandwich has flat surfaces and a few textures. For the object composed of flat planes, it is difficult to select correct correspondences because similar features exist on each side. Thus, accurate object pose estimation is required for objects composed of flat planes.

In this paper, object pose estimation method using color 2D code is proposed for the object composed of planes. In this method, the color 2D code is attached to the known

¹Graduate School of Engineering, Osaka City University, 3-3-138 Sugimoto Sumiyoshi-ku, Osaka, 558-8585 Japan {miyake@kdel., takubo@, ueno}@info.eng.osaka-cu.ac.jp
position on the product. The 2D code is usually used to know the embedded information and its pose from a captured image. However, the estimation of position using 2D code has some problems. If the 2D code is captured in front of a camera, the position can be calculated with high accuracy, but the accuracy of the 2D code’s position and orientation decreases as the captured angle is tilted. To solve this error, the ideal shooting position for each 2D code should be estimated, but it takes long time. In this study, 2D code is used only to get the embedded information, and the color of 2D code is chosen to stand out on the attached surface for adding the color feature. The attached color 2D code is aimed at acquiring the specific color feature easily to know the particular plane. To leverage the color feature, CSHOT descriptor is employed. CSHOT descriptor is an expanded version of SHOT descriptor, and it can express a histogram including shape and color information around a key point. The effectiveness of the proposed method is demonstrated using sandwich as an example that is difficult to recognize since it is composed of planes whose textures have few features. For the corresponding feature matching, 3D-HV (Hough Voting)[6] which returns acceptable inlier searching performance in real-time, and RANSAC (Random Sample Consensus)[13] which is known as the high precision performance are implemented and compared. Furthermore, 2.5D and entire 3D shape model are prepared to compare the computational time and the accuracy of the pose estimation to validate the feasibility of the real-time application.

II. OBJECT POSE ESTIMATION WITH 3D POINT CLOUD AND COLOR 2D CODE

The process of the proposed object pose estimation is illustrated in Fig. 1. In this paper, the target object is a sandwich, and its model is generated in the simulator by measuring with a depth camera. First, color 2D codes are attached to each surface of the sandwich. In order to set color feature at specific positions and easily obtain the characteristics of the shape, it is necessary to establish the attached place of color 2D code that can be easily attached to a specific place on each object. Therefore, we determine the position of the color 2D codes to the corners and edges of the object since it is easy and precise for attaching it. Each color of the code on the surface is different from each other to distinguish each surface of the sandwich by the color. In this paper, the color 2D code is not directly used to estimate the object pose, because we focus on the effectiveness of the intentional color information attachment to the object surface. The simultaneous use of 2D code and correspondence grouping algorithm is future work.

A reference model data is prepared by capturing 3D point cloud data using a depth camera in advance. When a scene is measured by the depth camera, the point cloud data of the scene and the reference model are compared to find the position of the target object. In the preprocessing, the unnecessary points: a floor and outlier point data, are detected and removed from the scene data to decrease computation time. Next, the features of both reference point cloud and scene point cloud are calculated by using CSHOT descriptor. As for the point-to-point feature correspondences between reference model and scene, we employ two methods: 3D Hough Voting and RANSAC. In Hough Voting, correspondences are determined based on the Euclidean distance using kd-tree, and clustering is performed by computing the set of Local Reference Flame (LRF) explicitly. In RANSAC, correspondences are chosen between some best matches at random, and clustering is performed by eliminating bad poses based on pose-invariant geometric consistencies of the inter-distances.

The calculation of CSHOT feature is described and demonstrated in Fig. 2. CSHOT feature has SHOT features as shape feature. SHOT descriptor encodes the normal vector of the points within a support sphere by histogram. The support sphere is divided into 32 sections resulting from 8 azimuth divisions, 2 elevation divisions and 2 radial divisions. Each angle (θ) between the normal vector of the key point (\( \vec{r} \)) and the normal vector of the point in the support sphere (\( \vec{n} \)) is calculated, and the histogram representing the distribution of angles is calculated in each division. In each local histogram,
the distribution of the vector cosine similarity ($\cos \theta$), which take a value from 0 to 1, is divided into 11 bins.

$$\cos \theta = \vec{n} \cdot \vec{r}$$ (1)

Thus, SHOT descriptor has 352 dimensions. CSHOT feature also has color feature. The L1 norm of the color triplets between on the keypoint and each point within 32 sections is calculated and binned into histograms per 31 bins. Therefore, CSHOT descriptor has 1344 dimensions.

In this paper, the program of pose estimation is implemented using Point Cloud Library (PCL). 2.5D reference model is prepared to compare to the calculation cost using the entire 3D reference model. The 2.5D reference model is assumed low calculation cost, which is used in the PCL tutorial, but not all pose of the target object can be detected since 2.5D model doesn’t have points on the back. On the other hand, the entire 3D shape model contains the entire shape data, but in the case of an object mainly composed of planes, it is assumed that many misrecognitions occur because priority is given to matching the planes.

III. EXPERIMENT

The objective of these experiments is to verify whether the target object pose can be estimated with desired precision by matching features with reference model. To show the effectiveness of the color 2D code attachments, the comparison experiments between using SHOT descriptor and CSHOT descriptor are presented. The reference model using 2.5D and entire 3D are also compared as for the computational time and the estimated position accuracy. Furthermore, two corresponding algorithms: 3D Hough Voting and RANSAC, are compared.

A. Experimental setup

A depth camera and a sandwich are arranged on dynamic simulator Gazebo as shown in Fig. 3. The sandwich on simulation is a triangular prism mapped by a sandwich’s photo as texture. In the scene, we estimate the sandwich pose rotated by every $\theta = 10$ deg from $\theta = -170$ deg to $\theta = 180$ deg. In this experiment with 2.5D model, the point cloud captured at $\theta = 20$ deg is used as a reference model as shown in Fig. 4. The simulator is executed on Ubuntu16.04, and the simulation PC has Corei7-7799HQ 3.8GH CPU, 16GB memory and 4GB GPU memory, but GPU acceleration of PCL is not used in this paper. The radius of the shot descriptor is set 2cm. Since the descriptor is divided into two in the radial direction, the acceptable translation error is set 1.00cm, and the acceptable angle error is set 4.00 deg in each axis corresponding to the object size.

B. Experimental results with 2.5D model

Table I shows the experimental results with SHOT descriptor, with CSHOT descriptor, and with CSHOT descriptor and color 2D code. The values in these tables represent in order of object pose in scene, distance between actual position in scene and detected position, the rotation error around x-axis, the rotation error around y-axis, and the rotation error around z-axis. The smaller error is, it indicates that the more precisely the object pose can be estimated. Bold fonts indicate precise estimation with error both within the acceptable error (1.00 cm and 4.00 deg). The values of $\theta$ not listed in these tables are omitted because the precision judgments of these values are not within desired accuracy.

From Table I, it is possible to accurately estimate 5 scene poses by using SHOT descriptor, 8 scene poses by using CSHOT descriptor, and 12 scene poses by using CSHOT descriptor and color 2D code with the desired precision. It is confirmed that the use of CSHOT descriptor with color 2D code is the best result for accurate pose estimation. The average computation time is within 1.0 to 2.0 sec.

Typical experimental results are shown in Fig. 5 and Fig. 6. Fig. 5 is typical false result using SHOT descriptor. Fig. 6 shows a success result using CSHOT descriptor and color 2D code in the case of SHOT descriptor failing. The sandwich on the left side is the reference model of $\theta = 20$ deg. The sandwich on the right side is the point cloud of $\theta = -20$ deg in search scene. The red point cloud represents the result of estimation by feature matching. The green lines represent sets of feature correspondences by matching.
impossible to estimate the pose of $\theta$ of the object data cannot be recognized. For example, it is was shown.

Therefore, from these results, the effectiveness of using CSHOT descriptor and color 2D code in feature matching can be avoided since this feature matching considers color information by using CSHOT descriptor and color 2D code. From this result, wrong estimation in search scene and estimated point cloud of sandwich are matched correctly. From this result, it is considered that SHOT descriptor represents by normal vectors only positional relationship of points, namely shape. On the other hand, as shown in Fig. 6, the point cloud of sandwich in search scene and estimated point cloud of sandwich are matched correctly. From this result, wrong estimation can be avoided since this feature matching considers color information by using CSHOT descriptor and color 2D code. Therefore, from these results, the effectiveness of using CSHOT descriptor and color 2D code in feature matching was shown.

When the 2.5D reference model is used, the back side of the object data cannot be recognized. For example, it is impossible to estimate the pose of $\theta = -160$ deg in scene with the reference model of $\theta = 20$ deg. To improve this, some reference models which are captured from multiple point of view can be prepared, but it is also impossible to judge which model is correct. It is necessary to use a 3D model for practical use.

As shown in Fig. 5, the point cloud of sandwich in search scene and estimated point cloud of sandwich are matched on two different surfaces. From this result, it is considered that SHOT descriptor represents by normal vectors only positional relationship of points, namely shape. On the other hand, as shown in Fig. 6, the point cloud of sandwich in search scene and estimated point cloud of sandwich are matched correctly. From this result, wrong estimation can be avoided since this feature matching considers color information by using CSHOT descriptor and color 2D code. Therefore, from these results, the effectiveness of using CSHOT descriptor and color 2D code in feature matching was shown.

C. Experimental results with entire 3D shape model

The entire 3D shape model is made by combining some 2.5D models. First, five 2.5D models, the point clouds of pose of $\theta = 0, 90, 180, -90$ deg and bottom, are prepared. In order to create the entire 3D shape model based on the point cloud of pose of $\theta = 0$ deg, the other point clouds in five are rotated and translated to fit the pose of $\theta = 0$ deg. Thus, we can prepare the entire 3D shape model as shown in Fig. 8.

Table II shows the experimental results with SHOT descriptor, with CSHOT descriptor, and with CSHOT descriptor and color 2D code. The bold characters indicate the precise estimation result whose thresholds are 1.00 cm distance and 4.00 deg angle errors. We can see the desired estimations; 1 scene pose by using SHOT descriptor, 12 scene poses by using CSHOT descriptor, and 13 scene poses by using CSHOT descriptor and color 2D code. Therefore, it is the best result among the 3 cases for accurate pose estimation when using CSHOT descriptor and color 2D code. It also found that the error of each scene is not symmetric around 0 deg of scene pose, regardless of the symmetric shape. The reasons are assumed that the texture of the sandwich is not

<table>
<thead>
<tr>
<th>scene $\theta$ (deg)</th>
<th>distance (cm)</th>
<th>x axis (deg)</th>
<th>y axis (deg)</th>
<th>z axis (deg)</th>
<th>distance (cm)</th>
<th>x axis (deg)</th>
<th>y axis (deg)</th>
<th>z axis (deg)</th>
<th>distance (cm)</th>
<th>x axis (deg)</th>
<th>y axis (deg)</th>
<th>z axis (deg)</th>
</tr>
</thead>
<tbody>
<tr>
<td>-30</td>
<td>1.99</td>
<td>156.74</td>
<td>37.93</td>
<td>68.61</td>
<td>0.98</td>
<td>156.37</td>
<td>54.33</td>
<td>71.85</td>
<td>0.55</td>
<td>10.54</td>
<td>6.10</td>
<td>3.95</td>
</tr>
<tr>
<td>-20</td>
<td>0.29</td>
<td>152.44</td>
<td>55.57</td>
<td>54.73</td>
<td>1.22</td>
<td>17.38</td>
<td>4.37</td>
<td>10.74</td>
<td>0.04</td>
<td>1.77</td>
<td>0.66</td>
<td>0.91</td>
</tr>
<tr>
<td>-10</td>
<td>1.86</td>
<td>160.56</td>
<td>63.17</td>
<td>32.06</td>
<td>2.80</td>
<td>32.58</td>
<td>0.46</td>
<td>21.57</td>
<td>0.48</td>
<td>0.22</td>
<td>0.04</td>
<td>0.14</td>
</tr>
<tr>
<td>0</td>
<td>0.17</td>
<td>5.47</td>
<td>0.18</td>
<td>3.55</td>
<td>0.26</td>
<td>1.68</td>
<td>0.02</td>
<td>1.07</td>
<td>0.14</td>
<td>0.16</td>
<td>0.07</td>
<td>0.07</td>
</tr>
<tr>
<td>10</td>
<td>0.68</td>
<td>1.64</td>
<td>0.32</td>
<td>1.10</td>
<td>0.09</td>
<td>0.44</td>
<td>0.08</td>
<td>0.29</td>
<td>0.17</td>
<td>0.41</td>
<td>0.02</td>
<td>0.26</td>
</tr>
<tr>
<td>20</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>30</td>
<td>0.35</td>
<td>1.89</td>
<td>0.27</td>
<td>0.99</td>
<td>0.15</td>
<td>1.30</td>
<td>0.69</td>
<td>0.89</td>
<td>0.02</td>
<td>0.09</td>
<td>0.31</td>
<td>1.48</td>
</tr>
<tr>
<td>40</td>
<td>0.28</td>
<td>1.54</td>
<td>2.60</td>
<td>1.62</td>
<td>0.74</td>
<td>1.18</td>
<td>0.80</td>
<td>0.77</td>
<td>0.02</td>
<td>0.21</td>
<td>0.03</td>
<td>0.01</td>
</tr>
<tr>
<td>50</td>
<td>2.01</td>
<td>158.56</td>
<td>49.54</td>
<td>8.15</td>
<td>0.21</td>
<td>0.09</td>
<td>0.08</td>
<td>0.06</td>
<td>0.05</td>
<td>0.19</td>
<td>0.19</td>
<td>0.67</td>
</tr>
<tr>
<td>60</td>
<td>1.75</td>
<td>158.65</td>
<td>44.93</td>
<td>31.47</td>
<td>0.24</td>
<td>0.20</td>
<td>0.18</td>
<td>0.10</td>
<td>0.17</td>
<td>0.39</td>
<td>0.65</td>
<td>0.12</td>
</tr>
<tr>
<td>70</td>
<td>0.24</td>
<td>3.38</td>
<td>3.73</td>
<td>0.02</td>
<td>0.24</td>
<td>0.29</td>
<td>1.71</td>
<td>2.17</td>
<td>0.31</td>
<td>0.35</td>
<td>3.56</td>
<td>0.76</td>
</tr>
<tr>
<td>80</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>90</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Table I

Precision judgment by the error using 2.5D model by Hough Voting

As shown in Fig. 5, the point cloud for creating an entire 3D shape model is made by combining some 2.5D models. First, five 2.5D models, the point clouds of pose of $\theta = 0, 90, 180, -90$ deg and bottom, are prepared. In order to create the entire 3D shape model based on the point cloud of pose of $\theta = 0$ deg, the other point clouds in five are rotated and translated to fit the pose of $\theta = 0$ deg. Thus, we can prepare the entire 3D shape model as shown in Fig. 8.

Table II shows the experimental results with SHOT descriptor, with CSHOT descriptor, and with CSHOT descriptor and color 2D code. The bold characters indicate the precise estimation result whose thresholds are 1.00 cm distance and 4.00 deg angle errors. We can see the desired estimations; 1 scene pose by using SHOT descriptor, 12 scene poses by using CSHOT descriptor, and 13 scene poses by using CSHOT descriptor and color 2D code. Therefore, it is the best result among the 3 cases for accurate pose estimation when using CSHOT descriptor and color 2D code. It also found that the error of each scene is not symmetric around 0 deg of scene pose, regardless of the symmetric shape. The reasons are assumed that the texture of the sandwich is not
difficulties of correct feature matching.  

The 2.5D model does not have. It is presumed that with the entire 3D shape model including the points of back each feature of the 2.5D model and entire 3D shape model. From this result, it is assumed that the reason why the feature matching with the entire 3D model could not be estimated poses which have been estimated with the desired precision could be estimated with 2.5D model could not be estimated poses that have been estimated with the desired precision. From these experimental results, there are more scene symmetrical, or the attached code of the color and texture are different on the left and right.

Now, we compare the estimation result for the case of using a 2.5D model and the case of using an entire 3D shape model. From these experimental results, there are more scene poses which have been estimated with the desired precision by using an entire 3D shape model than by using a 2.5D model. However, there are some cases that the poses that could be estimated with 2.5D model could not be estimated with an entire 3D shape model. From this result, it is assumed that the reason why the feature matching with the entire 3D shape model is not success, is in the process of computing each feature of the 2.5D model and entire 3D shape model. SHOT features or CSHOT features are described within the support sphere around each keypoint. When the keypoint is near the corner of the sandwich, the features are computed with the entire 3D shape model including the points of back side that the 2.5D model does not have. It is presumed that this could make the difference of each features and bring difficulties of correct feature matching.

D. Experimental results with entire 3D shape model by RANSAC

In this section, object pose is estimated by RANSAC using an entire 3D shape model as a reference model. Table III shows 19 scene poses can be accurately estimated. Thus, the method using RANSAC give the best performance of pose estimation in comparison with any experiments in this paper using Hough Voting. Although [15] says that Hough Voting is an ultra efficient algorithm which returns acceptable inlier and RANSAC often achieves extremely low precisions for inputs with scarce inliers, RANSAC achieves good precisions in this experiment. It is assumed the color information have good effect for making inliers. When the feature is changed from CSHOT to SHOT, estimation error is large in every scene pose, and it is not possible to estimate the target pose accurately. The average computation time is within 3.5 to 4.5 sec, which is slow rather than the 3DHV. This computation time is too slow for practical use. To reduce this time, using GPU and reducing the number of measurement points to the extent that precision is not impaired are considered in the

![TABLE II](image-url)
near future.

From these experimental results, it is resulted that RANSAC using CSHOT features is very effective in precisely estimating pose of object composed of planes. In addition, from these results, it is possible to estimate accurately at a known location on an object is also effective for estimating the pose of the object, which has few texture information. In the comparison of the correspondence feature matching algorithm, RANSAC is more accurate than 3D-Hough Voting. In addition, object pose estimation using 2.5D model is more accurate than using entire 3D shape model within the range where the pose of reference model is visible, but it is necessary to know the surface that the camera is actually seeing. Furthermore, it is also necessary to prepare multiple 2.5D models for target object in any angle, thus, pose estimation using an entire 3D shape model as a reference model is desirable.

For future tasks, the feature computation method and feature matching method are improved in case of using an entire 3D shape. Simultaneous use of 2D code position estimation and correspondence grouping algorithm is needed to be established for the practical use on the robot competition.

ACKNOWLEDGMENT

This work was performed in the Cooperative Research Project of DAIHEN Corporation.

REFERENCES