Tracking an object-grabbing hand using occluded depth reconstruction

Woojin Cho* KAIST UVR lab.
Gabyong Park† KAIST UVR lab.
Woontack Woo‡ KAIST UVR lab.

ABSTRACT

We propose a method that is effective in tracking 3D hand poses occluded by a real object. Since existing model-based tracking methods rely only on observed images to estimate hand joints, tracking generally fails when the hand joints are largely invisible. This problem becomes more prevalent when the tracked hand is grabbing an object, as occlusion by the object makes it harder to find a proper correspondence between the hand model and observation. The proposed method utilizes the occluded part of the hand as additional information for model-based tracking. The occluded depth information is reconstructed according to the geometric of the object and model-based tracking is employed based on particle swarm optimization(PSO). We demonstrate that the reconstructed depth information improves the performance of tracking an object-grabbing hand.

Index Terms: Computing methodologies—Computer vision problems—Tracking

1 INTRODUCTION

Augmented Reality(AR) and Virtual Reality(VR) are important technologies for the next generation. In this AR/VR environment, the use of hands provides the most natural and realistic means of interaction. In order to develop a hand gesture interface, advances in 3D hand pose estimation technology is crucial. Recent studies in this field have shown that isolated hand poses can be estimated quite accurately by a learned network that uses a large number of data sets or by various model-based methods [9]. However, in most of the environments where hands are actually used, there are many situations where hands holding real objects. Therefore, tracking hands that interact with surrounding objects still remains a challenge.

The reasons for this challenge are as follows: First, information on the hand is lost due to the occlusions made by the object. Second, to compensate for the lack of information on the hand pose with the information of the object, it is necessary to additionally find the geometry or pose of the object. Almost all of the studies that aim to solve these problems segment the hand and object to start pose estimation. A different approach estimates hand and object positions as heat map forms by using neural net instead of color based segmentation or known object initialization [1]. However, this approach is essentially the same with others in that it also relies on the concept of separating the hand from the object and estimating the pose by using each piece of data.

Our proposed method improves the pose estimation result by reconstructing reasonable depth data in an empty space in the hand image where the object region is segmented. After segmenting the hand and the object in a given depth image, the area in which the object was in the hand depth image will lose the depth value. However, if we use the lost depth image even if it is clear that the hand is occluded by an object, it will show a significant error in the subsequent tracking process. Invisible depth information yields wrong correspondence because the model-based tracking estimates the pose by fitting the model to observed depth image. Therefore, we need to regenerate this wrong depth region, which can give us a hint that the hand pose is correct even if the part of the hand model is located in the occluded area behind the object.

After reconstruction, the Particle Swarm Optimization(PSO) algorithm is used in the optimization process to estimate the hand pose using the depth input. PSO is a method that generates multiple particles corresponding to multiple pose candidates and has been adopted to many hand tracking algorithms. The proposed method was evaluated through a qualitative experiment based on a sequence of various modes of hand-object interactions. The results of the experiment show that our method of reconstructing depth data improves the tracking performance in situations where the hand interacts with an object.

2 RELATED WORK

Various methods have been suggested for robust hand pose tracking for hands interacting with objects. Several constraints (multi-camera setting, glove-based solution, etc) have also been proposed to solve the problem. We will discuss only the marker-less single RGB-D camera based approach in view of the overheads and drawbacks of each condition. There are three types of such methods: model-based, learning-based, and hybrid.

The model-based approach, also known as the generative method, estimates the hand pose by optimizing the objective function between the 3D hand model and the input frame. Under the assumption that the type and shape of an object are known, Kyriazis et al. [2] proposed an ensemble of collaborative trackers for tracking multiple interacting objects and used a physical simulation for hypothesizing the state of several objects. Sridhar et al. [7] showed a 3D hand model represented by a Gaussian mixture model which allows fast pose estimation. To track hands in interaction with unknown objects, Panteleris et al. [5] proposed a PSO-based algorithm that combines hand tracking and object modeling techniques. These methods are advantageous in tracking generalized pose, but it highly relies on the final solution of the previous frame and the current frame input. Accordingly, performance is decreased if the previous pose solution is erroneous or if there is insufficient information on the input of the current frame.

The learning-based approach typically estimates 3D hand poses on a single frame with a trained model. During the last few decades, various neural networks have been developed and methods of learning the hands interacting with objects in various environments have been attempted. Romero et al. [6] presented an approach based on nearest neighbor search that uses object as contribution to the pose estimation in a contextual fashion. Recently, Mueller et al. [3] showed hand-object tracking in cluttered scenes which used two combined Convolutional Neural Networks(CNN). As in the various fields where neural networks were used, this approach also showed excellent performance for environments similar to the learned data, but was not generalized to untrained poses and environments.

The hybrid method is a combination of the model-based and learning-based approach, which utilizes the advantages of each

*e-mail: dnwls2416@kaist.ac.kr
†e-mail: gypark@kaist.ac.kr
‡e-mail: wwoo@kaist.ac.kr

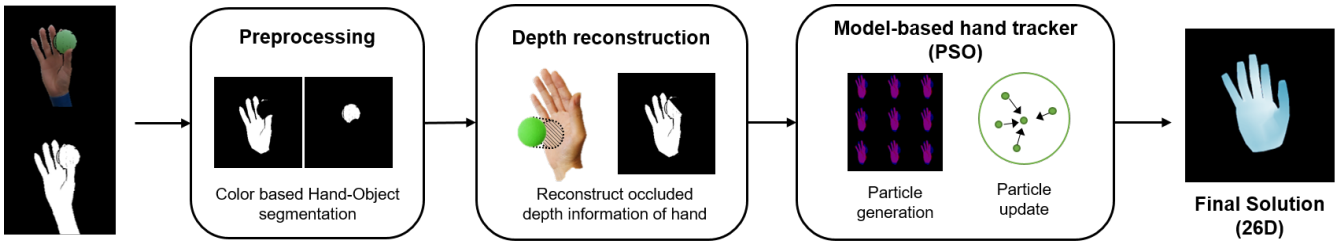

Figure 1: System Overview.

methods. Sridhar et al. [7] combined discriminative hand part classification with multiple proposal optimization. Methods of using interacting objects as constraints of the hand pose estimation process have also been proposed. Tzionas et al [8] proposed a framework that combines a generative model with discriminatively trained salient points. Chen [1] used that the grasp type of the hand can be inferred according to the shape of the object. The major benefit of using more than one method in various combinations is that it usually yields better results than using a single method. Still, the method is only estimating through observed images, the performance is lowered in highly occluded situations where there is lack of basis for estimation.

3 METHODOLOGY

Our system consists of the data processing and the model-based tracking. We assumed that the user wears a colored band on his wrist and the shape/color of the object are known. Figure 1 shows the pipeline of the proposed method. Input RGB-D images are obtained from an Intel Realsense SR300 sensor.

In the preprocess step, depth images were segmented into hand and object based on the color of the wrist band and the object represented in the YUV color space to reduce the influence of brightness. After segmentation, we reconstructed the depth for the intersection of the object area and the convex hull area of the hand. The reconstructed depth image was transferred to the tracker as input. Our tracker is based on the PSO algorithm which generates multiple hypothesis from a previous solution. Each hypothesis was represented with a tiled rendering technique similar to [4].

Hand Model : We adopted the parametric hand model with 26 DoFs(Degree of Freedom). 6 DoFs for a global 3D translation and a 3D rotation of model, 4 DoFs for a single finger. Each finger was represented by three joints, one saddle joint at the base which parameterized as two DoFs, and two hinge joints with one DoF. From 26 DoFs of a hand model, the depth map of the model was rendered and a discrepancy between the reconstructed depth image and the rendered depth image has been calculated. In every iteration of PSO, multiple hypotheses as the number of particles needed to be rendered. For this purpose, tiled rendering was performed through the OpenGL pipeline. Open source code from [2] was modified to generate the hand model.

Preprocessing : To segment given depth images to hand and object images, the RGB color space was converted to the YUV color space. As the Y component represents luminance information, we can minimize the influence of various light conditions by ignoring the Y component. More specifically, to recognize a colored wrist band, the U component was in range 70-116 and the V component was in 139-177 to recognize a colored wrist band. For object recognition, the U was in range 94-128 and the V was in 91-124.

Depth reconstruction

After segmenting hand with an object, the region where the object overlaps with the hand is taken as the target area for reconstruction. To determine which points are included in the target area, we used a convex hull of the segmented hand depth image.

Figure 2: Reconstruction area(Green).
RGB image(Left), Contour(middle), Convex hull(right).

As seen in figure 2, the outermost boundary points of a segmented hand depth image constitute an area containing almost all parts of the hand in most cases. We checked whether each 2D point corresponding to the object area exists in the convex hull area of the hand. If the point satisfied the condition, we calculated a proper depth value to reconstruct the point.

While 'proper depth value' can be defined in various ways, we assumed that the geometric information of the object is known in this study. By reconstructing the depth value corresponding to the back of the object, we supported the hypothesis that the hand model is placed behind the object. We used a spherical object for the experiment. In this case, the reconstructed depth value d_{recon} is defined as:

$$d_{recon} = d_{min} + 2r - gap \tag{1}$$

$$gap = \sum_{j=1}^{H} \sum_{i=1}^{W} D_{obj}(i,j) - d_{min} \tag{2}$$

where d_{min} is the minimum depth value in segmented object depth image (means that the depth of the closest point to camera), r is the radius of the sphere. gap represents the distance from the tangent plane of the closest point on the sphere to each point of the sphere surface, and D_{obj} represents the segmented depth image of the object. Through Eq. 1, we can calculate the depth value d_{recon} for the back of the sphere. For other types of objects, it can be done by accurately fitting the 3D object model and extracting the depth information of the backside of the 3D model.

Model-based Tracking

We solve the objective function for the model-based tracking. The goal is to estimate the hand pose parameters $x = \{x_1, x_2, ..., x_{26}\}$ in the following optimization problem:

$$\hat{x} = \min_{x \in R^{26}} E_d \tag{3}$$

where E_d is the fitting error between an observation and a rendered image.

We defined the objective function for the fitting error in a similar way to that in [4]. The computation involves pixel-wise calculations between the segmented depth image and the rendered depth image.

Figure 3: Self-Qualitative comparison. Each image set shows, RGB image(1st column), depth image(2nd column), estimated pose(3rd column). (a) Without depth reconstruction (b) With depth reconstruction

More specifically, the reconstructed depth image D_o and the rendered depth image D_r were compared in the objective function. The difference value was weighted by the value less than one because it is estimated rather than observed. The weight ω relies on whether the pixel value is reconstructed. Similarly, the normalized difference R between corresponding binary maps was calculated. A pixel in the binary map was set to one if the value exists in the depth image. When the model fits perfectly the segmented depth image in the binary map, the second term (1-R) becomes zero. The objective function E_d is defined as follows:

$$E_d = (1-R) + R\sum min(\omega|D_o - D_r|, d_M)/N \quad (4)$$

where d_M represents the value of the clamped depth difference and N is a normalization term.

To optimize the objective function, we employed a PSO in the model-based tracking. The PSO is a stochastic method that is performed by generating multiple candidate solutions (called particles) and iteratively updating those solutions. We set the number of particles to 64 and the number of generations to 30 for the model-based tracking. Multiple renderings and calculations of the Eq.4 are required in each generation. Since the process requires computationally large amounts, we performed optimized GPU calculations based on CUDA and OpenGL. Specifically, the proper distribution of GPU memory and a parallel reduction has been used along with a non-NULL stream execution has been used.

4 Experiment

Qualitative experiments were conducted to verify the effectiveness of the proposed method. We captured several sequences of the hand interacting with a spherical object. The sequences include actions such as rolling the ball around in the hand, holding it with fingers, and grabbing in various poses. While recording the sequence, we intended to occlude hand joints by the object as much as possible.

Figure 3 shows a qualitative comparison evaluation of the final hand model with and without the proposed method. Without the depth reconstruction process, the region corresponding to the object was removed as shown in the row (a), and an image with the depth value of 0 in the corresponding object area became an input of the tracker. When the depth reconstruction process was performed, the depth value corresponding to the back surface of the object was filled in the target area calculated with Eq. 1, as shown in row (b). In both cases, the tracker found a model pose with the lowest discrepancy error for given inputs. As a result, when the depth reconstruction was not performed, the hand joint information behind the object was missing in the transferred input, so that a completely incorrect pose result was obtained in a certain sequence. However, reconstructed depth values gave additional information about hand joints hidden by object, which yielded more realistic hand pose results. The computational time for the depth reconstruction process was measured to be less than 2.90 ms on average. This is a value that has less influence on real-time hand pose estimation.

Limitation

Figure 4: Failure cases due to incorrect depth reconstruction. (a) Without depth reconstruction (b) With depth reconstruction

The target area of the depth reconstruction was the intersection of the convex hull area of the hand and the object. Since there is no guarantee that the intersection area necessarily includes a finger behind, the wrong depth value may be reconstructed even though the actual finger is located elsewhere. Therefore, this approach is more effective in a situation where the object is held in the palm of the hand than when the object is placed between the fingers. Likewise, the proposed method adds a depth value of the back side of the object in the target area. However, a gap may exist between the back side of the object and where the actual hand is located. As seen in Figure 4, incorrect depth value led to an invalid hand pose as a result of optimization.

Our future work will obtain a confidence map for the reconstructed depth to apply an appropriate variable weight rather than the current fixed weight value. We will also extend the application of our depth reconstruction to objects with shapes other than spheres.

5 Conclusion

Accurate input data is imperative in estimating hand poses, regardless of whether the method is generative or discriminative. However, previous studies have shown that using separate depth data for the hand and object in situations where the hand is grabbing the object is not effective. It is difficult to obtain a good tracking performance by utilizing the depth input that has lost partial joint information located behind the object.

To solve this problem, we have supplemented a basis for hand joints hidden by an object through the reconstruction of depth information in that area. As a result of the PSO-based tracker, the proposed method provides more accurate poses in the hand-object grabbing sequences. Since the proposed method is deals with data processing in estimating hand pose, it can be easily applied to various existing researches. However, the proposed method may perform particularly well only when the hand is in contact with the object; otherwise, it may supplement wrong depth data. Therefore, we will perform a more effective depth reconstruction by applying object recognition and thereby enhancing the overall reliability of our results in the future.

Acknowledgments

This research was supported by Next-Generation Information Computing Development Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Science, ICT (NRF-2017M3C4A7066316).

References

[1] C. Choi, S. H. Yoon, C.-N. Chen, and K. Ramani. Robust hand pose estimation during the interaction with an unknown object. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3123–3132, 2017.

[2] N. Kyriazis and A. Argyros. Scalable 3d tracking of multiple interacting objects. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pp. 3430–3437. IEEE, 2014.

[3] F. Mueller, D. Mehta, O. Sotnychenko, S. Sridhar, D. Casas, and C. Theobalt. Real-time hand tracking under occlusion from an egocentric rgb-d sensor. In *Proceedings of International Conference on Computer Vision (ICCV)*, vol. 10, 2017.

[4] I. Oikonomidis, N. Kyriazis, and A. A. Argyros. Efficient model-based 3d tracking of hand articulations using kinect. In *BmVC*, vol. 1, p. 3, 2011.

[5] N. Plastira, V. Vouton, and H. GR70013. 3d tracking of human hands in interaction with unknown objects.

[6] J. Romero, H. Kjellström, and D. Kragic. Hands in action: real-time 3d reconstruction of hands in interaction with objects. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pp. 458–463. IEEE, 2010.

[7] S. Sridhar, F. Mueller, M. Zollhöfer, D. Casas, A. Oulasvirta, and C. Theobalt. Real-time joint tracking of a hand manipulating an object from rgb-d input. In *European Conference on Computer Vision*, pp. 294–310. Springer, 2016.

[8] D. Tzionas, L. Ballan, A. Srikantha, P. Aponte, M. Pollefeys, and J. Gall. Capturing hands in action using discriminative salient points and physics simulation. *International Journal of Computer Vision*, 118(2):172–193, 2016.

[9] S. Yuan, G. Garcia-Hernando, B. Stenger, G. Moon, J. Y. Chang, K. M. Lee, P. Molchanov, J. Kautz, S. Honari, L. Ge, et al. Depth-based 3d hand pose estimation: From current achievements to future goals. In *IEEE CVPR*, 2018.