Graph-cut-based 3D Model Segmentation for Articulated Object Reconstruction

Inkyu Han[*] Hyoungnyoun Kim[†] Ji-Hyung Park[‡]

Korea Institute of Science and Technology
University of Science and Technology

1 Introduction

The three-dimensional (3D) reconstruction of objects has been well studied in the literature of augmented reality (AR) [1, 2]. Most existing studies have assumed that the to-be-constructed target object is rigid, whereas objects in the real world can be dynamic or deformable. Therefore, AR systems are required to deal with non-rigid objects to be adaptive to environmental changes. In this paper, we address the problem of reconstructing articulated objects as a starting point for modeling deformable objects. An articulated object is composed of partially rigid components linked with joints. After building a mesh model of the object, the model is segmented into the components along their boundaries by a graph-cut-based approach that we propose.

2 System Overview

The processing sequence of our system is divided into three steps as shown in figure 1. In the first step, the target object is temporally reconstructed as a mesh model under the rigid motion assumption. The mesh model is built from a 3D surface mesh generation technique [1]. Through the approach, the model is created from 3D point features by a tetrahedralization and carving technique. The features are extracted from *keyframes*, which are unit frames representing differentiated sides of the object. To simplify the mesh building problem unlike in [1], we obtain the features from point clouds captured by a stereo camera. In the second step, which will be the main contribution of this paper, the reconstructed model from the previous step is segmented into individual rigid parts, and the joints of the object are extracted by articulated motion constraints [3]. In order to segment the model into partially rigid components along their boundaries, we apply a graph-cut algorithm given the graphical structure of the constructed model. As *articulated motions* of the object are detected, the link weights of the graph are updated by considering photometric and geometric information. In the last step, the partial components and the corresponding joint of the reconstructed model are tracked and augmented onto the scene by a multiple object tracking method [2].

3 Model Segmentation

The mesh model created via the model reconstruction step is split into two parts by using the max-flow algorithm [4]. A mesh model is a good source to generate a graph because their structures are almost similar. Vertices and edges of a mesh model become nodes and links of a graph, respectively. Only source and sink nodes are additionally inserted and linked to all other nodes. After a graph is generated from a mesh model, the user manipulates the object to show articulated motions, and links of the graph are weighted by the articulated motion constraints. Cutting process of the max-flow algorithm is performed whenever the links are updated, so that s/he can plan to show the as yet unsegmented side of the object by receiving the visual feedback.

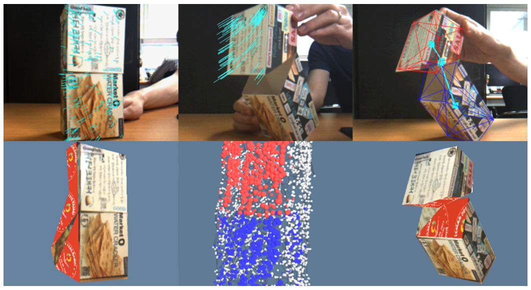


Figure 1: Three steps of our system: model reconstruction (left), model segmentation (center), and real-time tracking (right).

3.1 Articulated Motion Detection

Articulated motions of the object are used as cues to update the link capacities of the graph. If two components of the object are moved to different directions as a result of the user manipulation (see upper row of Figure 1), the angle of the joint between the components is changed. We postulate that an articulated motion is detected if the angle is sufficiently large. The articulated motions are detected based on a pose estimation technique [5] and RANSAC [6]. If an articulated motion is occurred, the estimated pose between a keyframe and an input frame is corresponded to only a component of the object. The features that are excluded as outliers by RANSAC belong to the other component of the object.

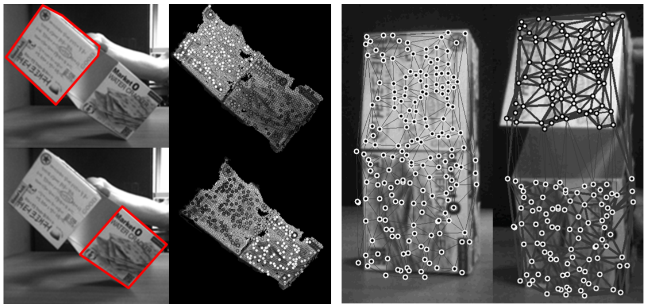
3.2 Graph Update

Poses of two components estimated from articulated motion detection are used to compute and update the capacities of the links in the graph structure. There are two kinds of links: *source-sink link* and n*eighbor link.* In the source-sink link, the nodes in a certain component are strongly connected with the source node, and the nodes in the other component are with the sink node. On the other hand the neighbor links between nodes in the same component are assigned larger capacities than those between nodes in different parts.

We measure the capacity of the source-sink link by combining photometric and geometric information. For the photometric measure, a template matching between a keyframe and an input frame is used. The 3D points of the features and the dense point cloud of the keyframe are projected onto each pose estimated from articulated motion detection. Subsequently, the matching scores of projected features f^{source}, f^{sink} for two poses are calculated between the projected point cloud and the input frame I_c by comparing local window centered on the feature. The matching

e-mail: {[*]ikhan, [†]nyoun, [‡]jhpark}@kist.re.kr

IEEE International Symposium on Mixed and Augmented Reality 2011
Science and Technolgy Proceedings
26 -29 October, Basel, Switzerland
978-1-4577-2185-4/10/$26.00 ©2011 IEEE



(a) (b)

Figure 2: (a) Link capacities of nodes connected to a source or sink node. Red rectangles indicate poses of two components. White and gray dots represent the capacities of source-sink links. (b) Capacity variation of neighbor links after articulated motions. The width of edges represents the strength of the capacity.

scores of f^{source} are used as the link capacities of the source, and the scores of f^{sink} are used for link of the sink (see Figure 2(a)). If the camera intrinsic matrix A is given, the capacities for projected features f_i of a k^{th} keyframe are defined by:

$$C_p(f_i^k) = TemplateMatching(f_i^k, I_c, \psi(A, P, M^k)) \qquad (1)$$

where $\psi(A,P,M) = APM$ is a function to project a point cloud M onto a certain pose P. Meanwhile, the Euclidean distances from a certain feature to two components are calculated for the geometric measure. Given the sets of inlier features i corresponded to two components, the geometric capacity for a feature f is defined by:

$$C_g(f_i^k) = \exp(-s_g \cdot d(f_i^k, \bar{i})) \qquad (2)$$

where $d(f, \bar{i})$ is the Euclidean distance between a projected feature f and a mean position $\bar{i}$ of a set of inliers, and s_g is a scaling factor. Finally, the capacities of source-sink links are calculated by weighted average of the photometric and geometric measures.

According to the articulated motion, the neighbor links are also determined. If two nodes are in a same component, the Euclidean distance of them remains unchanged even though articulated motions occur. On the contrary, the distance between nodes in different components is changeable (see Figure 2(b)). The distances among neighbor nodes are accumulated according to the articulated motions. Using the standard deviation of the accumulated data, the capacities of Neighbor links are calculated by Equation 3.

$$C_d(r_i^k, r_j^k) = \exp(-s_d \cdot \sigma(D(r_i^k, r_j^k))) \qquad (3)$$

where r_i is a feature point transformed from a keyframe, σ is the standard deviation, D is the set of Euclidean distances between two neighbor nodes, and s_d is a scaling factor.

Through the two types of capacities, nodes on the graph are split into two parts by using max-flow algorithm. After all nodes of the graph are segmented by changing a view for an input frame, the segmentation of the articulated object is completed.

4 Experimental Result

In our experiments, we used a Pointgrey Bumblebee2 Camera with a resolution of 640×480 to obtain images and 3D point clouds. The camera was set on a desk and objects were manipulated in front of the camera. Figure 3 shows some reconstructed models of three differently shaped objects with a single joint, and Figure 4 shows changes of link capacities according to articulated motions. We measured the capacities of four links connected to a certain node. As motions with an entirely distinguishable pair of poses were detected, the capacities of two links connected with both source node and a neighbor node in different part continuously decreased.

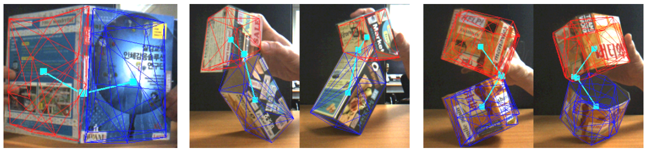


Figure 3: Reconstruction results of three objects.

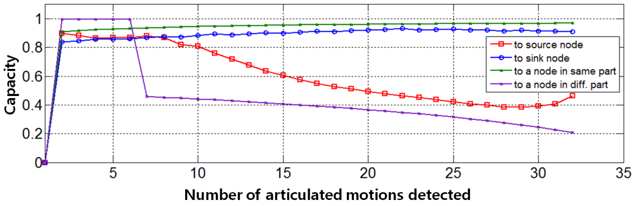


Figure 4: Capacity variation of links connected to a node.

5 Conclusion

In this paper, we presented a method for segmenting a mesh model of an articulated object from sequential stereo images. Coupled with an existing modeling and joint recovery techniques, our method reconstructed objects with various shapes successfully. We achieved robustness for our segmentation method by combining a graph-cut algorithm with proposed functions for updating link capacities. However, our research has limitations as follows. The quality of the reconstructed model is only guaranteed when feature points are evenly extracted from all sides of the object because the mesh generation and segmentation processes are performed based on feature matching. In addition, our approach considers objects composed of only two rigid parts linked by a joint. In the future, we plan to further investigate articulated objects with multiple parts and joints, and the issue of textureless object reconstruction is in our scope for the future plan.

Acknowledgements

This work was supported by the Global Frontier R&D Program on <Human-centered Interaction for Coexistence> funded by the National Research Foundation of Korea grant funded by the Korean Government(MEST) (NRF–M1AXA003–20100029752).

References

[1] Q. Pan, G. Reitmayr, and T. Drummond. ProFORMA: Probabilistic Feature-based On-line Rapid Model Acquisition. In *Proc. 20th British Machine Vision Conference (BMVC)*, 2009.

[2] K. Kim, V. Lepetit, and W. Woo. Keyframe-based Modeling and Tracking of Multiple 3D Objects. In *Proceedings of the International Symposium on Mixed and Augmented Reality*, 2010.

[3] G. K. M. Cheung, S. Baker, and T. Kanade. Shape-From-Silhouette of Articulated Objects and its Use for Human Body Kinematics Estimation and Motion Capture. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 16–22, 2003.

[4] Y. Boykov and V. Kolmogorov. An Experimental Comparison of Min-Cut/Max-Flow Algorithms for Energy Minimization in Vision. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9):1124–1137, 2004.

[5] R. Hartley and A. Zisserman. Multiple View Geometry in Computer Vision. *Cambridge University Press*, 2000.

[6] M. Fischler and R. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications ACM*, 24(6):381–395, 1981.