2015 IEEE International Symposium on Mixed and Augmented Reality

[POSTER] 2D-3D Co-segmentation for AR-based Remote Collaboration

Kuo-Chin Lien Benjamin Nuernberger Matthew Turk Tobias Höllerer
University of California, Santa Barbara
kuochin@ece.ucsb.edu, {bnuernberger, mturk, holl}@cs.ucsb.edu

Abstract

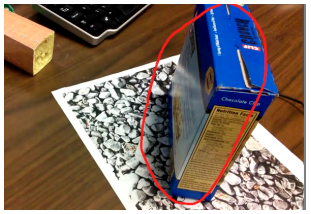
In Augmented Reality (AR) based remote collaboration, a remote user can draw a 2D annotation that emphasizes an object of interest to guide a local user accomplishing a task. This annotation is typically performed only once and then sticks to the selected object in the local user's view, independent of his or her camera movement. In this paper, we present an algorithm to segment the selected object, including its occluded surfaces, such that the 2D selection can be appropriately interpreted in 3D and rendered as a useful AR annotation even when the local user moves and significantly changes the viewpoint.

Index Terms: Human-centered computing [Human computer interaction (HCI)]: Interaction paradigms—Mixed / augmented reality

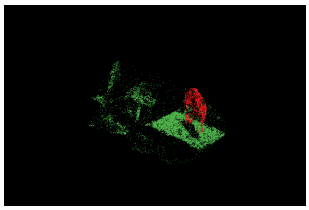
1 Introduction

AR-based remote collaboration systems such as [3] and [6] allow the remote user to draw 2D annotations to instruct the local user to accomplish a task (e.g., equipment repair or maintenance) that involves the physical environment. In a typical setup, the local user's camera's frames are wirelessly streamed over a network to the remote user where he/she can draw onto a frame in order to send an annotation back to the local user. These annotations must "stick" to the selected 3D object as the local user's camera moves, otherwise they will become useless or misleading. This is challenging for video-based object tracking algorithms since the object of interest can exhibit a large difference in appearance in different viewpoints. Gauglitz et al. [3] approached this problem by assuming a planar scene. To relax the planar scene assumption, they subsequently [4] proposed to incorporate an incrementally built Structure-from-Motion (SfM) 3D model of the unknown scene to infer the 3D positions of the 2D annotations. In particular, for selecting an object in the scene, they investigated several methods such as fitting a 3D plane to the points of a user's 2D stroke using the median depth of the stroke. These methods consider only the depth information of the stroke points but utilize neither the rich 2D image cues nor geometrical context of the 3D point clouds.

Similar to the work of Gauglitz et al. [4], we aim to consistently render the remote user's 2D annotation in every view when the local user moves his or her camera (e.g., an ellipse as shown in Figure 1(c)) using the sparse SfM point clouds constructed in the unknown scene. Unlike the planar annotation assumption of [4], however, we propose to take the input 2D annotation (e.g., Figure 1(a)) as a "user prior" and an additional 3D convexity prior to explicitly segment the object of interest in 3D, i.e., to label the 3D keypoints as foreground or background (Figure 1(b)). The idea of using convexity to help segmentation is rooted in psychophysical studies and has been reported in 2D interactive segmentation [5] as well as unsupervised 3D segmentation [10], but not in interactive 2D to 3D object selection.



(a) Remote user's annotation



(b) 2D-3D co-segmentation

(c) Ellipse sticks to the selected box

Figure 1: (a) The remote user draws a 2D ellipse to select the foreground box. (b) 2D-3D co-segmentation (colors: green for 3D points labeled as background, and red for those labeled foreground). (c) From a drastically different view point, the local user still can see the annotation correctly anchored on the 3D foreground object.

The segmentation problem investigated in this paper is also related to interactive multi-view image segmentation [7] but aims to obtain a good point cloud segmentation based on the annotation made in a single view. We refer to this as 2D-3D co-segmentation.

2 3D Point Cloud Segmentation Given 2D User Hint

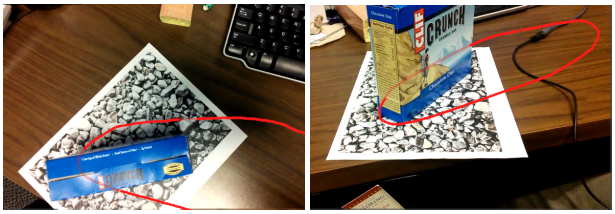
We formulate the interactive 2D-3D object co-segmentation problem as minimizing the following energy function:

$$E = E_{2D}(x, T^{-1}(y)) + E_{3D}(T(x), y), \quad (1)$$

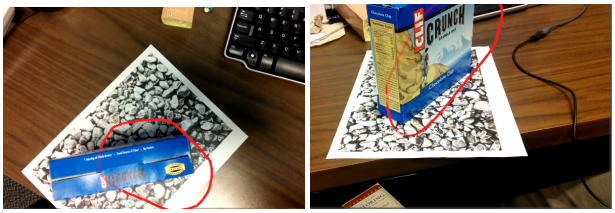
where x are the 2D points in the remote user-annotated frame I and y are the 3D points in the SfM model. All x and y are to be labeled as foreground or background in the optimization. T is a transformation that projects y to the image plane of the annotation, and T^{-1} projects x back to 3D. E_{2D} is a traditional 2D object segmentation energy, e.g., $E_{2D} = E_u(i) + E_p(i, j)$, where E_u aims to separate the foreground and background appearances (e.g., color distributions) and E_p is used to encourage neighboring points i and j to take the same label. E_{3D} is a convexity-based term to encourage the user selection to be propagated to a large convex hull, where the transition from convex to concave parts is more likely the separation point between objects.

We solve Equation (1) by a piecewise optimization strategy, i.e., iteratively solving one energy term and refining the solution using the other term. Solving the first term in Equation (1) is known to be NP-hard. Fortunately a user prior, such as a bounding ellipse, can give a good initial estimation of the foreground and background color distributions so that an expectation-maximization-style algorithm can solve it efficiently. Solving the second term requires checking the convexity of every potential foreground labeling configuration and assigning a cost accordingly, which is computation-

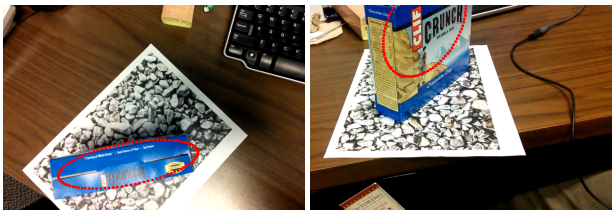
978-1-4673-7660-0/15 $31.00 © 2015 IEEE
DOI 10.1109/ISMAR.2015.56



(a) The dominant plane method [4]



(b) The median depth method [4]



(c) Our method: ellipse fitting the foreground point clouds

Figure 2: Two columns show two different AR views of the annotation interpretation results for the input annotation shown in Figure 1(a).

ally expensive. We use the method of Stein et al. [10] directly, which achieves 15 fps in our experiments due to a hard threshold being applied to reject potential 3D foreground configurations without strong enough convexity. More precisely, a region growing strategy is applied to propagate the foreground label, and the penalty of a potential labeling is set to be infinity if the convexity is not higher than a threshold. In other words, the foreground region stops growing toward the particular 3D point that incurs an infinite penalty.

In practice, we first fit an ellipse to the remote user's input drawing using the method described by Fitzgibbon and Fisher [2]. With this ellipse, we obtain the initial distributions of the depth values of the background as well as foreground for the use of the first term of Equation 1. For simplicity, only one iteration is performed. The next section summarizes the results of our preliminary experiments.

3 Results

Figure 2 compares the proposed method and the so-called "dominant plane" [4] and "median depth" [4] methods. In (a), the desk is identified as the dominant plane so all of the stroke points are interpreted as painted on the desk. In (b) the median depth of the stroke is assigned to all stroke points. One can see that both (a) and (b) rely on only the depth information on the strokes and thus mistakenly select background regions as foreground. The proposed method explicitly identifies the foreground points and can therefore correctly render the bounding ellipse on the box in any view.

4 Limitations and Ongoing Work

We are planning to build a dataset for a more thorough study on the 2D-3D object co-segmentation problem. There are few such resources available according to a very recent survey from the robotics community [1]. The most relevant one is the Object Segmentation Dataset [8], but its point clouds are constructed by a depth sensor from a fixed viewpoint and have quite different properties to our sparsely constructed SfM point clouds mainly captured surrounding an object of interest. For the same reason, the large body of RGBD datasets used in the computer vision community (e.g., [9]) are not directly suitable in our target AR application.

With this new dataset, we will investigate three key aspects of the 2D-3D object selection problem:

- Robustness. While man-made objects are often convex and can be extracted using a convex prior as reported in the segmentation literature [10] and as observed in our preliminary experiments, it is not clear yet how well the algorithm may work for more complex objects, e.g., a paper box like the one shown in Figure 1, but squashed.
- Scalability. Solving the segmentation currently takes seconds for point clouds with tens of thousands of 3D key points and a WVGA resolution input image, with the 2D segmentation the current bottleneck. Plus, given the fact that the 3D keypoints are incrementally added as the local user scans the object and environment, the computational load on solving the 3D energy term will also increase. More investigation is needed on how dense a point cloud can be without leading to prohibitively long computation times for our application and how coarse a point cloud might be to nullify the convexity-based object inference.
- User experience. With a manually labeled dataset, the objective mis-classification rate of a segmentation algorithm can be computed. But more important is the user's subjective evaluation of the algorithm, since with an abstract annotation hint, such as the ellipse, a user may not care about small segmentation errors. In addition, as mentioned in [4], users may have multiple ways to select a 3D object in the given 2D view. We believe that user-centered experiments with a comprehensive dataset will help to better understand general user behavior and preferences and thus advance the design of 2D-3D co-segmentation algorithms.

References

[1] B. Calli, A. Walsman, A. Singh, S. Srinivasa, P. Abbeel, and A. M. Dollar. Benchmarking in manipulation research: The YCB object and model set and benchmarking protocols. *arXiv:1502.03143*, 2015.

[2] A. W. Fitzgibbon and R. B. Fisher. A buyer's guide to conic fitting. In *the 6th British Conference on Machine Vision (BMVC 95)*, 1995.

[3] S. Gauglitz, C. Lee, M. Turk, and T. Höllerer. Integrating the physical environment into mobile remote collaboration. In *ACM MobileHCI*, 2012.

[4] S. Gauglitz, B. Nuernberger, M. Turk, and T. Höllerer. In touch with the remote world: Remote collaboration with augmented reality drawings and virtual navigation. In *The ACM Symposium on Virtual Reality Software and Technology (VRST '14)*, 2014.

[5] L. Gorelick, O. Veksler, Y. Boykov, and C. Nieuwenhuis. Convexity shape prior for segmentation. In *European Conference on Computer Vision (ECCV)*, 2014.

[6] S. Kim, G. A. Lee, and N. Sakata. Comparing pointing and drawing for remote collaboration. In *IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, 2013.

[7] A. Kowdle, Y.-J. Chang, D. Batra, and T. Chen. Scribble based interactive 3d reconstruction via scene co-segmentation. In *the IEEE International Conference on Image Processing*, 2011.

[8] A. Richtsfeld, T. Morwald, J. Prankl, M. Zillich, and M. Vincze. Segmentation of unknown objects in indoor environments. In *IEEE International Conference on Intelligent Robots and Systems (IROS)*, 2012.

[9] N. Silberman, P. Kohli, D. Hoiem, and R. Fergus. Indoor segmentation and support inference from rgbd images. In *European Conference on Computer Vision (ECCV)*, 2012.

[10] S. C. Stein, F. Worgotter, J. P. Markus Schoeler, and T. Kulvicius. Convexity based object partitioning for robot applications. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2014.