SCRIBBLE BASED INTERACTIVE 3D RECONSTRUCTION VIA SCENE CO-SEGMENTATION

Adarsh Kowdle$^1$, Yao-Jen Chang$^1$, Dhruv Batra$^2$, Tsuhan Chen$^1$

$^1$Cornell University, NY, USA. $^2$Toyota Technological Institute, Chicago, USA.

ABSTRACT

In this paper, we present a novel interactive 3D reconstruction algorithm which renders a planar reconstruction of the scene. We consider a scenario where the user has taken a few images of a scene from multiple poses. The goal is to obtain a dense and visually pleasing reconstruction of the scene, including non-planar objects. Using simple user interactions in the form of scribbles indicating the surfaces in the scene, we develop an idea of 3D scribbles to propagate scene geometry across multiple views and perform co-segmentation of all the images into the different surfaces and non-planar objects in the scene. We show that this allows us to render a complete and pleasing reconstruction of the scene along with a volumetric rendering of the non-planar objects. We demonstrate the effectiveness of our algorithm on both outdoor and indoor scenes including the ability to handle featureless surfaces.

Index Terms— image based modeling, interactive 3D reconstruction

1. INTRODUCTION

We consider a scenario where the user has taken a few images of a scene from varied poses. The goal is to obtain a visually pleasing reconstruction of the scene. Automatic algorithms such as [1, 2, 3], etc have been shown to work well with a large collection of images. When the input images are restricted, these automatic algorithms fail to produce a dense plausible reconstruction. There are a number of multiview stereo algorithms which try to obtain a dense depth map for the scene from a set of images [4]. However, multiview stereo algorithms are known to be slow and with a small set of images the reconstruction is usually incomplete, leaving holes on textureless surfaces and specular reflections. In order to improve the reconstruction, some algorithms make planar approximations to the scene [5, 6]. This allows for more visually pleasing reconstructions. However, these algorithms use features such as strong edges and lines which may be absent in textureless surfaces or non-planar objects (like walls, trees, people, etc). This has led to interactive algorithms.

Prior interactive reconstruction algorithms, require involved user-interactions ranging from providing feature correspondence, to marking edges, plane boundaries and detailed line models of the scene [7, 8, 9]. In this paper, we present a novel scribble based interactive 3D reconstruction algorithm where we relax the interactions to mere scribbles and render a planar reconstruction of the scene. In a typical scene, non-planar objects occluding the scene can result in holes in the scene reconstruction. The strength of our approach is the ability to use surface-level correspondence across multiple views to create composite texture maps for the scene thereby rendering pleasing planar reconstructions of the scene. Moreover, we use this correspondence to obtain volumetric reconstructions of the occluding object. All of this involves very simple interactions in the form of scribbles.

Scribbles have been used in interactive algorithms in the past. They were first used by Boykov et al. for interactive segmentation to indicate foreground and background [10]. Batra et al. used scribbles for interactive cosegmentation [11], which was applied to object of interest 3D modeling by Kowdle et al. [12]. Srivastava et al. improve the 3D model obtained using their Make3D algorithm by using scribbles to enforce coplanarity in their MRF formulation [13]. Sinha et al. used scribbles for interactive cosegmentation [11]. We use scribbles from the user to indicate the surfaces and non-planar objects in the scene. We believe we are the first to use these scribbles in a multi-class segmentation framework.

An overview of our algorithm is illustrated in Fig.1. Our algorithm allows the user to pick any image and provide simple scribbles to indicate planar surfaces and non-planar objects in the scene. We use the scribbles to learn an appearance model for each surface and then, formulate the multi-class segmentation task as an energy minimization problem over superpixels, solved via graph-cuts. This scene segmentation along with the sparse 3D point cloud from structure-from-motion (SFM) helps define the geometry of the scene. We introduce an idea of 3D scribbles which helps propagate this scene geometry to the other images to co-segment the images into the various planes and objects in the scene. The scene co-segmentation helps obtain a composite texture map for the scene eliminating holes due to occluding objects, giving a pleasing planar reconstruction of the scene. In addition to this, we use the co-segmentation of non-planar objects in the scene to obtain a visual hull for the occluding object [12], which is rendered as part of the prior planar reconstruction of the scene. We now describe our approach in detail.
perpixels, with edges between adjacent superpixels. The colon (\(\cdot\)) is used for penalizing label disagreement between neighbors. The second term (smootheness term) is used for penalizing label disagreement between neighbors.

\[ E_i(X_i : S) = \sum_{i \in \mathcal{V}} E_i(X_i : A) + \lambda \sum_{(i,j) \in \mathcal{E}} E_{ij}(X_i, X_j), \quad (1) \]

where the first term (data term) indicates the cost of assigning a superpixel to one of the labels, while the second term (smoothness term) is used for penalizing label disagreement between neighbors. The colon (\(\cdot\)) in the equation indicates that the term is dependent on the learnt appearance model.

**Data (Unary) Term.** Our appearance model consists of a Gaussian Mixture Model for each of the \(p\) surfaces labeled, i.e., \(A = \{\text{GMMS}_1, \ldots, \text{GMMS}_p\}\). Specifically, we use colour features extracted from superpixels [12] on the labeled sites and fit GMMS for the corresponding classes. The data terms for all sites are then defined as the negative log-likelihood of the features given the class model. We set the unary term of the superpixels labeled by the user to \(-\infty\) (a large negative value) as hard constraints in the energy minimization.

\[ E_{ij}(X_i, X_j) = I(X_i \neq X_j) \exp(-\beta), \quad (2) \]

where \(I(\cdot)\) is an indicator function.

Finally, we use graph-cuts (with \(\alpha\)-expansion) to compute the MAP labels for all superpixels, using the implementation by Bagon [15] and Boykov et al. [16, 17, 18]. The result segments the image into the different surfaces labeled by the user as shown in Fig.2(a); we call this scene segmentation.

**2.3. 3D scribbles and scene co-segmentation**

Image co-segmentation has gained a lot of popularity in the community [20, 21, 11]. However, co-segmentation of the multiple surfaces in the scene is not as trivial as the two class image co-segmentation since, it is hard to define features discriminative between geometric surfaces. However, when a user provides scribbles on an image, they are doing so based on their perception of the geometry of the scene, i.e. they are
not just indicating surfaces and objects in that image but, are giving us cues about the 3D scene geometry common across all the images. This is the common thread between the images we exploit to perform the co-segmentation.

**3D scribbles.** Using the estimated plane parameters and the camera projection matrix of the scribbled image, we develop the idea of 3D scribbles. Let the projection matrix of camera $i$ be defined as $M_i = K_i R_i (I - C_i)$ where, $K_i$ is the intrinsic matrix, $R_i$ is the rotation matrix and $C_i$ is the camera center in the world co-ordinate system. Consider, a 2D scribble point $s_{1,j}$ seen from $Cam_1$, on a segment which corresponds to the plane $l$ parameterized by $[\hat{n}_l \cdot d_l]$ where, $\hat{n}_l$ is the plane normal and $d_l$ is the plane constant. The projection of this scribble point on another image seen from $Cam_2 (s_{2,j})$ is given by,

$$s_{2,j} = K_2 R_2 \left( \frac{(-d_l - \hat{n}_l \cdot C_1)}{\hat{n}_l \cdot [K_1 R_1]^{-1} s_{1,j} + C_1} - C_2 \right)$$

We take care to avoid warping the scribbles onto occluded planes by using the scene geometry and camera pose. For example, we can eliminate many of the warped scribbles by considering only the planes visible from a particular view.

**Scene co-segmentation.** The resulting scribbles on all the images are as shown in Fig.2(c). Using these scribbles as hard constraints on all the images, we now extend the energy minimization based multi-class labeling described in Sec.2.1 to all the images thereby achieving co-segmentation of all the images into the multiple scene classes Fig.2(d).

### 2.4. Visualization

We develop a back-projection algorithm using the equation above, to evaluate the point of intersection of a ray from the camera center through every pixel on the image plane, and the estimated 3D surface. Using these 3D points, we generate a mesh for the scene with the corresponding image texture and render a texture mapped planar reconstruction of the scene as shown in Fig.1(c), enabling pleasing fly-throughs.

### 2.5. Rendering non-planar objects

The algorithm described so far renders a planar reconstruction of the scene. In case of non-planar objects, we get in input from the user to indicate these objects in the scene, as shown in the blue ellipse in Fig.5(c). This tells the algorithm which surface corresponds to the non-planar object. We then estimate an approximate planar proxy for the object, which helps position the object as part of the rendered scene.

**Object co-segmentation.** At this stage, the algorithm knows which surface indicated by the user corresponds to the non-planar object. We treat the scribbles corresponding to the non-planar object as foreground scribbles and all other scribbles as background scribbles and use ideas from prior work by Kowdle et al. [12] to obtain a 3D visual hull of the non-planar object via a 2-class co-segmentation, which is rendered using an independent mesh. The scene co-segmentation also allows us to create a composite texture map for the scene covering up holes due to occlusions as shown in Fig.3(a).

Once the algorithm generates the 3D reconstruction, the user can provide more scribbles to indicate new or previously occluded planes, and improve the result, thus closing the loop on our interactive 3D reconstruction algorithm.

### 3. RESULTS AND DISCUSSIONS

We test our algorithm on a number of scenes (both indoor and outdoor) rendering pleasing, complete reconstructions. Fig.1(d) shows the result on a scene with featureless surfaces, Fig.4 how more results on such planar scenes. To show the power of this algorithm to render non-planar objects we show the results with the tree in the outdoor scene in Fig.3(b) and the person in the indoor scene in Fig.4(d). Please see video summary\(^3\) with fly-through of the 3D reconstructions.

**Comparison.** We compare our results with other approaches which one can use to reconstruct a scene. Using SFM [1], on

---

\(^3\)Video Summary: [http://chenlab.ece.cornell.edu/projects/Interactive_3D](http://chenlab.ece.cornell.edu/projects/Interactive_3D)
a huge image collection can render dense point clouds however, with the scenario we consider, the point cloud is very sparse and far from dense. Multi-view stereo algorithms like patch-based multi-view stereo (PMVS) render a denser reconstruction. However, this fails to render a complete reconstruction, leaving holes in the presence of textureless surfaces and specular surfaces. As we show in Fig.6, the results from our interactive reconstruction algorithm is more complete.

Fig. 6. Comparison with patch-based multi-view stereo: The top images show the reconstruction generated by PMVS with the errors shown in black ellipses, while bottom images show our results with corrected reconstructions shown in blue ellipses (Best in color).

To compare our approach with other interactive works, we show our result on the play-house dataset of Sinha et al. [9], in Fig.4(c). We achieve a good reconstruction with very limited interactions. We note that prior works require tedious user interactions to mark the planes in the scene or provide line models of the scene, while we achieve good results using simple scribbles to indicate the surfaces. Moreover, the planar modeling used in the prior works does not allow for reconstructing non-planar objects like the tree, however our approach allows making the reconstruction more complete.

4. CONCLUSIONS

In this paper, we present a novel interactive 3D reconstruction algorithm which uses simple user interactions in the form of scribbles to indicate the surfaces and non-planar objects in the scene. We introduce the idea of 3D scribbles to propagate the scene geometry and co-segment the scene across multiple views. We render a planar reconstruction of the scene and introduce the idea of overlaying a volumetric rendering of the occluding non-planar object as part of the scene thus rendering a more complete reconstruction of the scene.

5. REFERENCES


4We use the PMVS implementation by Furukawa et al. [2] and available at http://gradient.cs.washington.edu/software/pmvs/pmvs-1/index.html