CORRECTING 3D SCENES ESTIMATED FROM SETS OF MULTI-VIEW IMAGES USING SHAPE-FROM-COONTOURS

Matthew Grum and Adrian G. Bors

Department of Computer Science, University of York, York YO10 5GH, UK
Email: adrian.bors@york.ac.uk

ABSTRACT

This paper proposes enforcing the consistency with segmented contours when modelling scenes with multiple objects from multi-view images. A certain rough initialization of the 3D scene is assumed to be available and in the case of multiple objects inconsistencies are expected. In the proposed shape-from-contours approach images are segmented and back-projections of segmented contours are used for enforcing the consistency of the segmented contours with 3D objects from the scene. We provide a study for the physical requirements for detecting occlusions when reconstructing 3-D scenes with multiple objects.

Index Terms— 3D scene modelling, multi-view images, occlusions, image segmentation

1. INTRODUCTION

3D scene modelling from multi-view images has lately received a significant interest from the research community due to its many applications. Single object reconstruction has attracted considerable research interest by using various approaches such as multi-view stereo, shape-from-silhouettes [1, 2, 3], shape-from-shading [4], etc. Various 3D object representations are used for the scene modelling including voxels [5], radial basis functions (RBF) [6] and meshes of vertices [7]. Space carving assigns voxels to a 3D object or carves them away from its volume according to their photoco

2. 3D SCENE CORRECTION USING SHAPE-FROM-COONTOURS

Let us assume that we have a set of images \( I = \{I_i| i = 1, \ldots, n\} \), each taken from a different viewing direction, which are characterized by their projection matrices \( P = \{P_i| i = 1, \ldots, n\} \). In the following we assume that we have an initial approximate representation of a scene \( S \) with multiple objects. In the experimental results we consider space carving [5] followed by implicit RBF modelling [6] for the initialization stage. The approach using image disparities in the projections of 3D patches [9] relies on good textures but does not provide suitable results in areas of uniform colour. Nevertheless, such areas in the images can be segmented well.

Let us assume that the scene contains at least two distinct objects \( \{A, B\} \in S \) closely located to each other. We consider that each object outline from the 3D scene is projected onto segmented contours, of their corresponding projections, in the input images, denoted as \( \{a_i, b_i\} \in I_i, i = 1, \ldots, n \) where \( a_i = P_iA \) and \( b_i = P_iB \) and \( P_i \) represents the projection matrix from the 3D scene to the \( i \)th image. In this paper we consider image segmentation for defining the contours of objects, such as \( \{A, B\} \) in 3D or \( \{a_i, b_i\} \) in the 2D images from \( I \). By comparing the projections of the segmented objects in 3D with their corresponding image segmentations we can detect any inconsistencies and use them to correct the 3D scene, \( S \).

In the following we consider both unsupervised and supervised image segmentation for extracting object contours. We consider a feature space for each pixel represented by a vector \( z = \{r, g, b, \zeta\}^T \), containing the three colour components and a texture feature \( t \), weighted by \( \zeta \), provided by the density in the output of the Harris corner detector. For the unsupervised image segmentation we use the mean shift algorithm [11], which was employed for image segmentation in [12]. Pixels with features closest to each of the local maxima, in the given feature space, are assigned to the corresponding segmented regions. For the supervised segmentation we use support vector machines (SVM) [13].

The initial estimates of the object boundaries are refined by using active contours such as snakes, in order to join edge maps, representing boundaries of segmented objects from images, into contours of objects \( C_i \). The consistency of the object contours, resulting after projecting the 3D object outlines \( C_i \) is verified against those corresponding to segmented object contours from images. The inconsistencies between \( P_iC_i \) and the contours \( C_i \), extracted from each image are detected and then used to correct the 3D scene. The visual hull, denoted by \( H_i \), represents the outer bound of the scene based on its appearance in several images and was used for modelling single
3D objects in the shape-from-silhouettes algorithm [1, 2, 3]. The concept of the visual hull is applied in our approach to individual objects from the scene considering the segmentation of its projections in images from several view-points. In this case, the visual hull of objects, denoted as \( \mathcal{H}_c(\mathbf{a}_i) \) and \( \mathcal{H}_c(\mathbf{b}_j) \) is provided by the object contours such as \( \mathbf{a}_i \) or \( \mathbf{b}_j \) from each image \( i = 1, \ldots, n \), where these objects are visible. After comparing sets of pixels corresponding to 2D contours and those from the projected 3D contours we identify the regions corresponding to the difference sets as:

\[
\{ \mathbf{c} \mid \mathbf{P}_i\mathbf{c} = (\mathbf{S(\mathbf{P}_i\mathbf{C}_{i})}) \cup (\mathbf{S(\mathbf{C}_{i})}) \} \]

where \( \mathbf{S(\mathbf{C}_{i})} \) represents the set of pixels located in the interior of contour \( \mathbf{C}_{i} \), \( i = 1, \ldots, n \) and \( \mathbf{c} \in \mathcal{S}_i \) represents a point from the 3D scene, whose projection \( \mathbf{P}_i\mathbf{c} \) lies among the pixels from the area between the sets \( \mathbf{P}_i\mathbf{C}_{i} \) and \( \mathbf{C}_{i} \) from each image. Points located in the regions from the 3D scene after the back-projection of the set from (1) are displaced to their nearest surface in 3D along its surface normal. Eventually, such points would be located in \( \mathbf{S(\mathbf{P}_i\mathbf{C}_{i})} \cap \mathbf{S(\mathbf{C}_{i})} \) ensuring the consistency of \( \mathcal{H}_c(\mathbf{a}_i) \) and \( \mathcal{H}_c(\mathbf{b}_j) \) with the 3D scene \( \mathcal{S}_i \). This methodology can be applied to various surface representations such as voxels, parametric (including RBFs) and meshes, where surface self-intersections must be avoided [7]. The scheme describing the main stages for the entire methodology for 3D scene modelling is shown in Fig. 1 and was described in detail in [14].

3. ANALYSIS OF OBJECT SEPARABILITY WHEN RECONSTRUCTING SCENES WITH MULTIPLE OBJECTS

This section discusses the case when two distinct objects are fused together in a single larger shape in the attempt to reconstruct the 3D scene. Such situations arise due to various image factors including object occlusion, uncertainty in the camera parameters, various illumination conditions, image noise, etc. [7, 9]. In the following we consider a simple artificial scene consisting of two identical objects located in the center on a flat plane, observed by a number of surrounding cameras. In order to evaluate the degree to which the shape influences the merging error, two different object shapes are considered: a pair of cuboids with square bases of side length \( l \) and height \( 2l \) and a pair of cylinders of diameter \( l \) and height \( 2l \). The objects stand upright on an horizontal plane separated by a gap of size \( d \) (the distance between object centers is \( d + l \)), as shown in Fig. 2 for the two cuboids. There are two scenarios, one in which the two objects are distinct, and another in which they are joined by filling in the gap between them, as it could result after errors in the 3D scene reconstruction. This study aims to identify whether two objects are merged or not.

![Fig. 1: Diagram showing the 3D scene reconstruction stages.](image)

In the following we consider two error measures, assessing the differences between pairs of contours and the projected surface differences, respectively. The first error measure is the Hausdorff distance, [15], which is defined as:

\[
e_{H}(F,G) = \max_{\mathbf{C}_F} \min_{\mathbf{g} \in \mathcal{C}_G} || \mathbf{f} - \mathbf{g} ||
\]

where \( \mathcal{C}_F \) and \( \mathcal{C}_G \) represent the contours of the projections corresponding to the fused objects and to the separate objects, respectively, while \( \mathbf{f} \) and \( \mathbf{g} \) are points located on these two contours. The area error is evaluated as the difference between the surfaces closed by the two contours:

\[
e_A(F,G) = \frac{|| \mathbf{S(\mathbf{F})} \cup \mathbf{S(\mathbf{G})} \setminus (\mathbf{S(\mathbf{F})} \cap \mathbf{S(\mathbf{G})}) ||}{|| \mathbf{S(\mathbf{G})} ||}
\]

where \( \mathbf{S(\mathbf{F})} \) and \( \mathbf{S(\mathbf{G})} \) are the areas that correspond to the projections of the 3D objects in the hypotheses of fused and separated objects, respectively, and \( \cdot \) represents the cardinality.

Figs. 3a and 3b show the plots of the area error \( e_A(F,G) \) with respect to camera angles.

Fig. 2: Scene representing two cuboids and the parameters

A scene is characterized by six extrinsic camera parameters, three of which correspond to the location of the center of projection and three to the angles which determine the camera orientation. The camera positions in space can be expressed in spherical coordinates by two angles and the distance from the scene center, modelled by \( \phi, \theta, r \), as shown in Fig. 2. In order to simplify the scene assumptions we fix \( r \), the distance from the center of the scene, as well as the elevation angle \( \theta \), while we vary the azimuth angle, \( \phi \). All cameras are assumed to be pointing towards the center of the scene with no tilt/roll. 2D projections of these scenes are obtained using the intrinsic camera parameters, thus at least geometrically resembling the images you would get from a real system though they lack image noise, illumination variation, shadows, etc.

![Fig. 3: Area error \( e_A(F,G) \) with respect to camera angles.](image)
peak and thus to the ability to differentiate separate and joined objects when varying θ, until $\theta = \frac{3\pi}{4}$, when the error starts to increase for all values of $\phi$. The most extreme case is considered for $\theta = \frac{\pi}{2}$. In this configuration, the gap is always visible as the camera is located above the objects and thus the variation with the azimuth angle $\phi$ no longer has any effect. The availability of high elevation images is relatively easy to achieve for a collection of small objects but more difficult for scenes consisting of urban landscapes with buildings.

Fig. 5: Surface error $e_{A}(F, G)$ variation with number of cameras.

In the following, $d$ is varied by moving the objects away from each other as well as from the origin which is located on the support plane at the mid-distance between the two objects. A circular configuration of cameras is considered at the elevation corresponding to $\theta = 0.85$ radians. There are differences between the errors for various configurations of evenly spaced $n$ cameras, with different azimuth angle $\phi$ offsets. For $n$, we record the minimum and maximum area errors $e_{A}(F, G)$, measured between the fused and separate case hypotheses when considering all possible offsets. The results representing the best and worst cases for detecting the separation for each configuration are shown in Fig. 5 when increasing the number of cameras used, starting from $n = 2$. The best case results are almost totally invariant of $n$ for evenly spaced cameras. Two cameras can be sufficient if they happen to be in exactly the right position in order to observe the gap between the objects. When increasing the number of cameras, the likelihood of detecting the gap between the objects increases as well, until $n = 20$, when the worst case performance curve converges asymptotically towards the better circumstances situation.

Fig. 6: Six images from the image set displaying multiple objects.

4. EXPERIMENTAL RESULTS

In the following we present the results when reconstructing scenes with multiple objects using multi-view images. Six images, from a larger set of $n = 12$ images of a scene showing five objects captured from various viewpoints, are displayed in Figs. 6a-f. We initialize the 3D scene using space carving [5], represent its surface with implicit RBFs [6], and correct the image disparities from projections of 3D patches as in [9]. The resulting 3D scene is shown in Fig. 7a. It can be observed that two of the objects representing a knife-block and a kettle are merged together as shown in the closer view from Fig. 7b. The result provided by the shape-from-silhouettes (SFS) [1, 2] when applied on the original set of 12 images is shown in Fig. 8a. In Fig. 8b we provide the result when SFS is applied onto the 3D scene from Fig. 7a. We apply the shape-from-contours (SFC). The main difference between SFC and SFS occurs in the regions where two or more objects are located nearby or may be touching each other. The condition when two objects are wrongly merged has been analysed in Section 3. Object contours are extracted from the image set and compared with the contours resulting from projecting the current 3D scene onto the image planes using identical camera parameters $P$. 

Fig. 4: Errors evaluated when increasing the inter-object distance $d$.

Fig. 5: Surface error $e_{A}(F, G)$ variation with number of cameras.
with those of the original images. Both unsupervised and supervised image segmentation are considered for object contours extraction. Mean shift clustering [11, 12] and SVM [13] are used for the unsupervised and supervised segmentation, respectively. Segmentation results for the two merged objects are shown in blue in four different views in Figs. 9a-d and 9e-h, when using unsupervised and supervised segmentation, while the projection of the fused object surface from Fig. 7b, is shown in red. We can observe a large discrepancy in Figs. 9c and 9g and a smaller one in Figs. 9a, 9b, 9e and 9f. The regions between the two contours, displayed using red and blue, are back-projected into the 3D scene and their corresponding volumes corrected. The corrected 3D scene, following unsupervised segmentation, is shown in Fig. 12. The middle of the scene is now visible through the gap between the kettle and knife-block. These results are better than those provided by SFS directly on the original image set from Figs. 8a and 8b and they improve the 3D reconstruction of the area between the kettle and knife block when compared with those from Fig. 7.

(a) Initial 3D scene   (b) Segmented fused objects
Fig. 7: 3D scene representations using implicit RBFs.

(a) Applied on the initial image set   (b) Applied on 3D initial estimate
Fig. 8: Shape-from-silhouettes results.

(a) (b) (c) (d)
(e) (f) (g) (h)
Fig. 9: Projections of the 3D scene onto image planes, shown with red, and object contours resulting from image segmentation, shown with blue, using mean-shift in (a)-(d) and SVM in (e)-(h).

Numerical errors are evaluated for assessing the improvement in the 3D scene when using SFC from either unsupervised or supervised image segmentations. Numerical results are provided in

(a) Hausdorff distance   (b) Area difference
Fig. 10: Numerical accuracy evaluation.

Figs. 10a and 10b when varying the azimuth angle φ, for the Hausdorff distance [15] and the area error $e_A(F, G)$ from (3) as defined in Section 3. An error peak, which corresponds to the region located between the kettle and the knife-block, can be identified in all four curves from the two plots in Fig 10. We also evaluate the PSNR, for a region selected inside the black rectangle from Fig. 11, between the original image and the corresponding projections from the corrected 3D scene. For this region, the initial PSNR of 11.17 dB is improved to 13.91 dB and to 13.75 dB, respectively, when using unsupervised and supervised segmentations for SFC. Six different views of the entire 3D scene reconstruction are shown in Fig. 12, considering the same projection parameters $P$ as for the original images shown in Fig. 6, after enforcing unsupervised SFC consistency.

(a) (b) (c) (d)
Fig. 11: Selected region.

Fig. 12: 3D scene corrected using unsupervised SFC.

5. CONCLUSION
This paper proposes shape-from-contours (SFC) method for estimating scenes of multiple objects, assuming an approximate initial estimation of the 3D scene. SFC uses back-projections of segmented contours of objects from images in order to improve the 3D scene with multiple objects. Both supervised and unsupervised segmentation are used for segmenting given multi-view images. A study is provided for detecting merged objects depending on the geometry of the scene as well as the necessary number of cameras in order to detect errors due to multi-object occlusions in scenes. The proposed methodology can be applied for correcting and modelling 3D scene representations for 3D data coding from multi-view images.
6. REFERENCES


