Object Recognition and Localization while Tracking and Mapping

Robert O. Castle* David W. Murray†

Active Vision Laboratory, Department of Engineering Science, University of Oxford, UK

ABSTRACT

This paper demonstrates how objects can be recognized, reconstructed, and localized within a 3D map, using observations and matching of SIFT features in keyframes. The keyframes arise as part of a frame-rate process of parallel camera tracking and mapping, in which the keyframe camera poses and 3D map points are refined using bundle adjustment. The object reconstruction process runs independently, and in parallel to, the tracking and mapping processes. Detected objects are automatically labelled on the user's display using predefined annotations. The annotations are also used to highlight areas of interest upon the objects to the user.

Index Terms: H.5.1 [Multimedia Information Systems]: Artificial, augmented, and virtual realities—; I.4.8 [Scene Analysis]: Object Recognition—Tracking

1 INTRODUCTION

In this paper, we describe a method to allow users of a wearable augmented reality system to view AR constructs not only at locations of interest, but on objects of interest. A modification of Klein's parallel tracking and mapping (PTAM) method [4] that allows a user to create and traverse multiple maps is used for camera tracking and mapping [2]. The parallel tracking and multiple mapping system (PTAMM) is combined with a method based on feature descriptors to allow simultaneous recognition, reconstruction, and localization of objects within 3D maps.

The camera is tracked from frame-to-frame over the short term, and its pose, along with the positions of the 3D map points, optimally recovered at keyframes using bundle adjustment. At the same time, known planar objects are detected and recognized using Lowe's SIFT descriptors [5], and are located by optimizing their 3D structure by triangulation, with the keyframe camera poses determined by bundle adjustment used as fixed quantities. The objects used here are planar, but planarity is imposed after, not during, reconstruction. There is almost no object modelling involved, with the method requiring only an image of the object to function.

2 OBJECT DETECTION IN KEYFRAMES

The object detection, recognition and localization process runs in a separate thread from the tracking and mapping threads, allowing it to be all but independent from the rest of the system. The process uses the keyframes from the mapping process to find objects, and its outputs are augmentations to the 3D map and do not influence the map's evolution.

2.1 Object database entries

To recognize objects a database of known objects is required. This is constructed using frontal images of the objects of interest. After correcting for radial distortion, SIFT keypoint descriptors $\boldsymbol{\sigma}^i$ and their positions $\boldsymbol{x}^i$ are computed following Lowe's method [5]. The j-th database entry becomes

$$\begin{aligned}\mathcal{O}_j \quad = \quad & \{\, \mathcal{I}_j, \quad \{\boldsymbol{\sigma}^i_j, \boldsymbol{x}^i_j\}_{i=1\ldots I_j}, \quad \{\boldsymbol{x}^m_{\mathrm{B}j}\}_{m=1\ldots M_j}, \\ & \{\boldsymbol{x}^n_{\mathrm{AR}j}, \text{“AR-markup”}\}_{n=1\ldots N_j} \,\}\end{aligned} \tag{1}$$

containing the frontal image $\mathcal{I}_j$ of the object, the list of SIFT descriptors $\boldsymbol{\sigma}^i_j$ and their image locations $\boldsymbol{x}^i_j$, the locations of several boundary points $\boldsymbol{x}^m_{\mathrm{B}j}$ to define the object extent, and lastly a number of positions $\boldsymbol{x}^n_{\mathrm{AR}j}$ tagged with graphical annotations.

2.2 Object detection and recognition

While running on-line, once a keyframe k has been selected for processing (described later), SIFT descriptors and their locations $(\boldsymbol{\sigma}^l_k, \boldsymbol{x}^l_k, l = 1\ldots L_k)$ are extracted from its associated image. These keypoints are stored in the keyframe structure.

The keypoint descriptors are compared with those in the database using Beis and Lowe's approximate best-bin-first modification of kd-trees [1], which provides faster look-up than linear search. If the number of keypoints matched between the keyframe image and any given object's database entry exceeds a threshold, that object is flagged as potentially visible. However, because of repeated structure or other scene similarity, some of the features may be incorrectly matched. Here the object's planarity is exploited to remove outliers, by using RANSAC to estimate the homography between the database feature positions and the keyframe feature positions, $\boldsymbol{x}_j = \mathtt{H}\boldsymbol{x}_k$, and inferring that the object is indeed visible if the consensus set of inliers is large enough. The inliers are added to a list of observations for their particular database keypoint, for use in the localization process. The homography itself is discarded.

2.3 Keyframe selection

To triangulate a keypoint in 3D it needs to be observed in at least two keyframes. As the recognition process runs independently from the mapping process any keyframe could be selected and processed.

To enable the most efficient processing of keyframes for timely presentation of information, keyframes are selected in pairs in the following manner: the first processed is that keyframe whose position and orientation are closest to the current camera's; and the second is the keyframe that is most similar to the first.

To assist the search for this pair, whenever a keyframe is added to the map, the keyframe in the map that is most similar is recorded. This becomes its parent, and the parent also records that this new keyframe is a child, forming a bidirectional tree. The similarity measure used is the number of map points the two keyframes have in common. This bidirectional tree allows all of the most similar frames to be quickly located for any particular frame.

3 OBJECT RECONSTRUCTION AND LOCALIZATION

Once an object has been found visible in two or more keyframes, there will be a subset of object keypoints that were observed in two or more of the keyframes. First their scene positions are reconstructed quite generally, and only then are they fitted to the underlying shape of the model to obtain the position and orientation of the object in the scene.

*e-mail: bob@robots.ox.ac.uk

†e-mail: dwm@robots.ox.ac.uk

3.1 Keypoint triangulation

A keypoint is triangulated by treating the keyframe poses as fixed, as PTAMM's bundle adjustment has already optimized them. With just two views the usual algebraic residual is minimized [3]. Up to scale, the two observations of the homogeneous scene point $\boldsymbol{X}$ are $\boldsymbol{x}_1 = \mathtt{P}_1\boldsymbol{X}$, and $\boldsymbol{x}_2 = \mathtt{P}_2\boldsymbol{X}$, where the projection matrix for each view $\mathtt{P}_{1,2} = \mathtt{K}[\mathtt{R}_{1,2}|\boldsymbol{t}_{1,2}]$ is known. Combining these,

$$\mathtt{A}\boldsymbol{X} = \begin{bmatrix} x_1\boldsymbol{p}_{13} - \boldsymbol{p}_{11} \\ y_1\boldsymbol{p}_{13} - \boldsymbol{p}_{12} \\ x_2\boldsymbol{p}_{23} - \boldsymbol{p}_{21} \\ y_2\boldsymbol{p}_{23} - \boldsymbol{p}_{22} \end{bmatrix} \boldsymbol{X} = \boldsymbol{0} \tag{2}$$

where $\boldsymbol{p}_{ij}$ is the j-th row of $\mathtt{P}_i$, and the residual is minimized when $\boldsymbol{X}$ is, up to scale, the column of $\mathtt{V}$ corresponding to the smallest singular value in the SVD $\mathtt{UDV}^\top \leftarrow \mathtt{A}$. As more observations are added, Levenberg-Marquardt (LM) is used to minimize error in the image, and the inhomogeneous $\boldsymbol{X}$ is estimated so as to minimize the L_2 norm of errors in the image

$$\boldsymbol{X} = \arg\min_{\boldsymbol{X}^*} \left\{ \sum_k ||\boldsymbol{x}_k - \boldsymbol{x}_\mathrm{p}\left(\boldsymbol{X}^*, \mathtt{P}_k\right)||_2 \right\} \tag{3}$$

where $\boldsymbol{x}_k$ is the (inhomogenous) observation in keyframe k and $\boldsymbol{x}_\mathrm{p}()$ is the predicted image position. When the map is adjusted the keyframes may move. This is handled by checking for a change in keyframe poses and rerunning LM for the affected objects.

3.2 Plane fitting

Once at least three keypoints have been localized a plane is fitted to them using RANSAC to expose outlying data. For the inlying set, the mean and covariance

$$\boldsymbol{\mu} = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{X}_i \qquad \mathtt{C} = \sum_{i=1}^{n} (\boldsymbol{X}_i - \boldsymbol{\mu})(\boldsymbol{X}_i - \boldsymbol{\mu})^\top \tag{4}$$

are computed. The plane normal $\hat{\boldsymbol{n}}$ is the column of $\mathtt{U}$ corresponding to the smallest eigenvalue in $\mathtt{\Lambda}$ from the eigendecomposition $\mathtt{U\Lambda U}^\top \leftarrow \mathtt{C}$. The inliers are now projected onto the plane,

$$\boldsymbol{X'}_i = \boldsymbol{X}_i - \hat{\boldsymbol{n}}(\hat{\boldsymbol{n}} \cdot (\boldsymbol{X}_i - \boldsymbol{\mu})), \tag{5}$$

and these in-plane point locations are now used to locate the object on the plane. The optimized locations of the keypoints found by the bundle adjustment process are left unchanged.

3.3 Object fitting

The final stage is to fit the database keypoint locations $\boldsymbol{x}^i$ to those on the located plane $\boldsymbol{X'}_i$. Then the boundary points of the object can be found in 3D.

For n iterations, where here $n = 100$, two of the projected inlier points $\boldsymbol{X'}_i$ are selected at random to act as scaling and rotation reference points. The database keypoints of the inlier set are transformed from the image plane of the object to the estimated 3D plane, relative to the two projected inlier points. The pair that result in the minimum distance between points are accepted as the best match. The boundary points $\boldsymbol{x}_\mathrm{B}$ are then found relative to the best pair of in-plane points, and saved as additional data with the map. The same is also done for any AR annotations $\boldsymbol{x}_\mathrm{AR}$ located on the object. The object can now be used in the AR rendering process.

4 Results

The system has been implemented in C++, and runs under Linux on a 2.20 GHz Intel Dual Core processor. It is used here to identify paintings in a gallery, using a database of 37 paintings with some 75 000 features. PTAMM's multiple-map capability is used, with separate maps made along each of three walls with paintings, and the relocalizer used to switch between them. 3D views of the three maps are shown Fig. 1. In each of these the detected paintings, keyframes, and the tree structure linking the keyframes is shown. As the maps are created, the paintings are detected, localized and labelled for the user with two examples shown in Fig. 2.

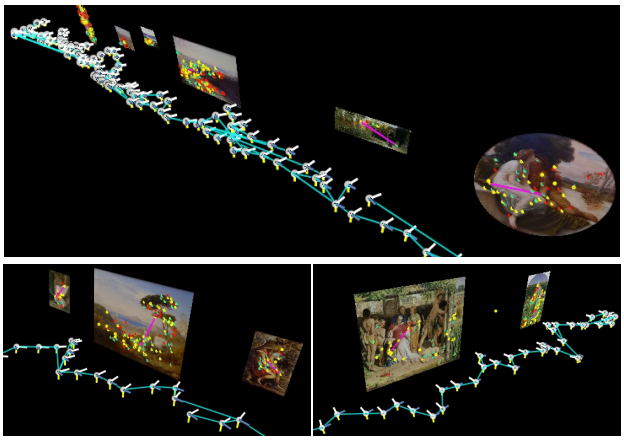

Figure 1: 11 paintings located within the 3 gallery maps. The bidirectional tree connecting the keyframes can also be seen.

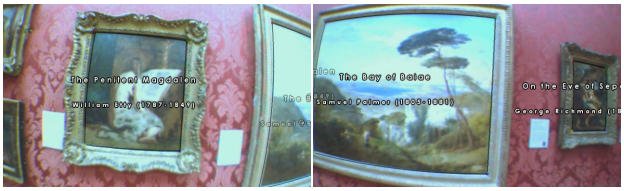

Figure 2: The detected paintings are automatically labelled.

5 Conclusion

This paper has shown how objects can be recognized and their shape reconstructed and localized within a 3D map using observation and matching of SIFT features between keyframes. Only a single image is required, greatly simplifying the modelling process. Using the dense and well spaced keyframes generated by the underlying mapping process allows the mapped environment to be thoroughly searched for known objects. The object detection process runs independently, and in parallel to, the mapping and tracking processes providing the 3D location of detected objects within a map. Automatic labelling of objects allows a user to freely explore an environment while being presented with relevant information.

Acknowledgements

This work was supported by the UK EPSRC (grant EP/D037077).

References

[1] J. S. Beis and D. G. Lowe. Shape indexing using approximate nearest-neighbour search in high-dimensional spaces. In *Proc IEEE Conf on Computer Vision and Pattern Recognition, Puerto Rico*, pages 1000–1006, 1997.

[2] R. O. Castle, G. Klein, and D. W. Murray. Video-rate localization in multiple maps for wearable augmented reality. In *Proc 12th IEEE Int Symp on Wearable Computing, Pittsburgh PA*, pages 15–22, 2008.

[3] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, second edition, 2004.

[4] G. Klein and D. W. Murray. Parallel tracking and mapping for small AR workspaces. In *Proc 6th IEEE/ACM Int Symp on Mixed and Augmented Reality, Nara, Japan*, 2007.

[5] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.