Visual Localization for Mobile Surveillance

Valeria Garro, Maurizio Galassi, Andrea Fusiello
Dipartimento di Informatica, Università di Verona

Abstract

In this paper we propose a complete system that is able to accurately localize a mobile agent wearing a camera inside a known environment. The work leverages on a precomputed 3D structure to obtain 2D-3D correspondences and then orients the camera. Experiments in a challenging environment with a handmade ground-truth demonstrate sufficient accuracy to support the target application on real scenarios.

1. System overview

Our system leverages on a structure-and-motion pipeline, called SAMANTHA [4], that produces a sparse set of 3D points endowed with appearance descriptors (the "model") by processing a unordered set of images of the scene (the "images archive").

Localization or *orientation* of the camera occurs via a linear algorithm that requires a set of 2D-3D point correspondences between the current frame and the model. Since typically the 2D points visible in one image are a small subset of the whole reconstruction, it is highly advisable to deploy pruning strategies to limit the set of 3D candidates. Our technique is based on retrieving the most similar images to the current frame from the archive and then limiting the candidates to those points that are visible in the retrieved images. Retrieval follows a standard Bag-of-words (BoW) approach with tf-idf weighting [6].

The system involves two main stages (see Fig.1):

- an "offline" stage that runs SAMANTHA and indexes images according to the BoW approach.
- an "online" stage during which the video stream captured from the mobile camera is transmitted over Wi-Fi connection to a server that processes each frame in order to orient the camera, thereby localizing the mobile agent wearing it.

In particular, the online stage consists of the following steps, as illustrated in Fig.1:

This work has been funded by the EU Project SAMURAI.

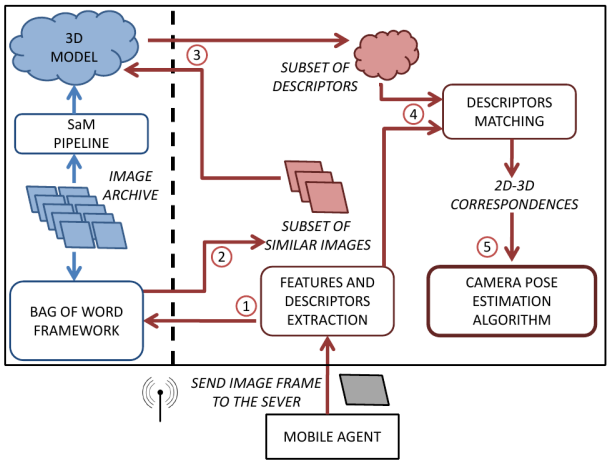

Figure 1. System Overview. The "offline" data pre-processing step are marked in blue, the "online" stages are drawn in red.

1. Fast-Hessian features detection and SURF descriptor extraction [1];
2. retrieval of the most similar images with BoW;
3. recover of SURF descriptors related to the 3D points viewed by the retrieved images;
4. descriptors matching (closest neighbors with check on the second-best match);
5. camera orientation (or pose estimation) from 2D-3D correspondences with Fiore's algorithm[2].

2. Experiments

We run our test on a challenging outdoor environment consisting of a parking space located in between several buildings with repetitive structures. We recorded a video sequence with a proprietary device specifically designed within the EU project SAMURAI.

To build the 3D model, 678 images (with resolution 2048×1536) of the whole scene has been taken with a consumer camera, sampling almost all the area every five meters. Four static calibrated cameras are located on the

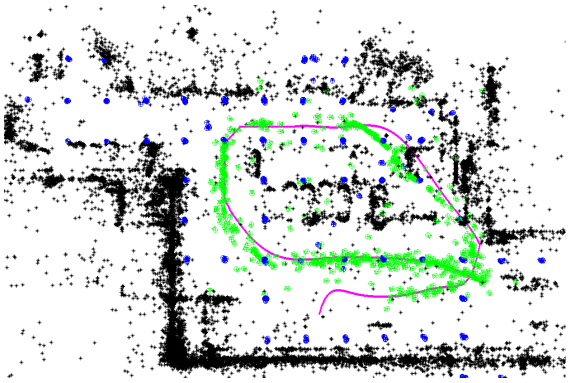

Figure 2. The ground-truth trajectory is drawn in magenta, the mobile cameras oriented by our algorithm are shown as green markers and the cameras reconstructed by SAMANTHA are marked in blue.

parking area corners. They are used to extract the ground truth trajectory of the mobile agent.

For the retrieval step a k-means clustering has been processed with $k = 6000$. Feature points have been extracted with 'Fast-Hessian' detector setting the threshold equal to 500. We tested two different camera orientation algorithms, Fiore's (in the formulation of [3]) and EPnP [5] a fast non-iterative solution whose implementation is available online.

Fig. 2 shows with green markers the positions of the mobile camera as produced by our retrieval-based algorithm (with Fiore's orientation).

Figure 3 shows the histogram of the distance of the localized cameras to the respective ground-truth, by applying respectively Fiore's and EPnP algorithms for the camera orientation. The histograms show that Fiore usually returns most of the samples in the first bins associated with low error values, meaning that accuracy is typically under one meter and outliers suddenly occur. On the other hand, EPnP orients more frames but with error spread in the higher part of the histogram with outliers that overpass 30 meters. The average distance errors are 2.86 m for Fiore and 2.97 m for EPnP. The mean accuracy of Fiore outperforms EPnP but the number of localized frames is higher using the latter algorithm, as shown in Tab. 1.

The performances of our current C++ implementation – running on a Intel QuadCore with 2.4Ghz – are reported in Table 2.

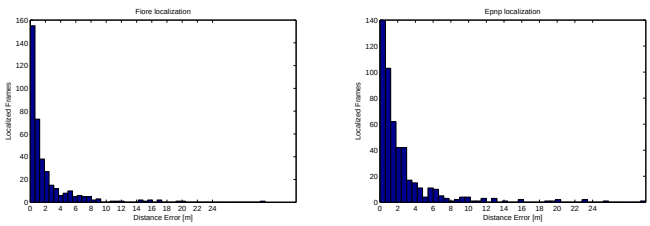

Figure 3. Histogram of distance error: Fiore algorithm (left) and EPnP algorithm (right).

Orientation algorithm	Average Error [m]	Localized frames
Fiore + MSAC	2.86	383/877
EPnP + MSAC	2.97	511/877

Table 1. Comparison between Fiore and EPnP algorithm: average error of camera localization and number of localized frames.

Steps	Times [sec]
Surf Extraction	0.55
Retrieval	0.35
Features Matching	0.65
Fiore + MSAC	0.18
EPnP + MSAC	0.19

Table 2. Average performance of our CPU implementation

References

[1] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded-up robust features (surf). *Computer Vision and Image Understanding*, 110(3):346–359, 2008. 1

[2] P. D. Fiore. Efficient linear solution of exterior orientation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(2):140–148, 2001. 1

[3] A. Fusiello. A matter of notation: Several uses of the Kronecker product in 3-D computer vision. *Pattern Recognition Letters*, 28(15):2127–2132, 2007. 2

[4] R. Gherardi, M. Farenzena, and A. Fusiello. Improving the efficiency of hierarchical structure-and-motion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1594 – 1600, San Francisco, CA, 13-18 June 2010. 1

[5] F. Moreno-Noguer, V. Lepetit, and P. Fua. Accurate non-iterative o(n) solution to the pnp problem. In *Proceeding of the IEEE International Conference on Computer Vision*, Rio de Janeiro, Brazil, October 2007. 2

[6] J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. In *Proceedings of the International Conference on Computer Vision*, volume 2, pages 1470–1477, Oct. 2003. 1

An extended version of this paper can be found on the web at: http://profs.sci.univr.it/~fusiello/papers/iwmv11-e.pdf