On Tracking and Matching in Vision Based Navigation

Adam Schmidt, Marek Kraft, and Michał Fularz(✉)

Institute of Control and Information Engineering,
Poznan University of Technology, Poznan, Poland
{adam.schmidt,marek.kraft,michal.fularz}@put.poznan.pl

Abstract. The paper presents a thorough comparative analysis of the feature tracking and the feature matching approaches applied to the visual navigation. The evaluation was performed on a synthetic dataset with perfect ground truth to assure maximum reliability of results. The presented results include the analysis of both the feature localization accuracy and the computational costs of different methods. Additionally, the distribution of the uncertainty of the features localization was analyzed and parametrized.

1 Introduction

Establishing point features correspondences across images in video sequence plays an important role in the visual navigation of robots. The correspondences can be used to estimate the transformations between the consecutive poses as in the visual odometry (VO) systems [1][2][3] or to update the environment model in the simultaneous localization and mapping (SLAM) [4][5][6].

The contemporary visual navigation systems use either the feature tracking or the feature matching approach. In the first case the position of the features on the new image is determined by finding the most probable displacement of the features within the local neighbourhood of their last positions. This approach is used in several visual navigation systems such as [3]. The more popular approach is based on finding the characteristic points on the analysed images, calculating the descriptor of their local neighbourhood and finding the pairs of the most similar descriptors. The examples of the systems using the feature matching paradigm include e.g. [5] and [7].

According to Fraundhofer and Scaramuzza [2] the tracking-based approach is usually more suited for small-scale environments and frame to frame tracking. The descriptor-based matching is generally used in larger environments where the displacement of the camera introduces significant changes to the features' local neighbourhood. In such cases the matching may be performed less frequently to compensate for the computational cost of the descriptor calculation and matching of the early descriptors such as SIFT [8] or SURF [9]. However, the introduction of the FAST detector [10] and binary descriptors such as BRIEF [11] or ORB [12] significantly changes this distinction.

A. Campilho and M. Kamel (Eds.): ICIAR 2014, Part II, LNCS 8815, pp. 461–468, 2014.
DOI: 10.1007/978-3-319-11755-3_51

Over the years, significant attention has been paid to the evaluation of different point features detectors and descriptors [13]. However, no similar research of the feature tracking algorithms has been performed. Moreover, to the extent of the authors knowledge, no study comparing the efficiency of the tracking and matching paradigms in the context of visual navigation is available.

This paper presents the evaluation of the feature tracking and matching approaches to the feature localization task. Moreover, the analysis of the features' localization uncertainty was performed. The synthetic, rendered data was used in the experiments guaranteeing the precision of the reference camera poses, which is especially important considering that the errors of features localization are measured in single pixels.

2 Methods

Feature matching and feature tracking are two alternative approaches to the problem of finding keypoint correspondences across a sequence of images. Feature tracking starts with finding points of interest in the initial image, and tracking them in consecutive frames by finding their correspondences using local search methods, e.g. correlation or gradient descent. Feature tracking performs best if the viewpoints in which the images were taken are not too far apart. Significant apparent feature motion caused by viewpoint change is usually associated with the deformation of the features' neighbourhood, making tracking significantly more prone to failure than matching.

Feature matching is based on direct keypoint-to-keypoint comparison rather than on local search. Each keypoint is assigned a unique descriptor computed based on the distribution of the image intensity function in the feature's neighbourhood. To match the features pairwise between consecutive images, a similarity metric is computed between their descriptors and the pairs with a smallest distance are considered to be matches. The descriptors are designed to cope with some degree of distortion of feature neighbourhood. This makes them better suited for finding feature correspondences whenever one has to deal with a wide baseline.

Historically, feature tracking had the advantage of being less computationally demanding, as the first robust detectors and descriptors were quite complicated both to calculate and to match [8][9]. With the advent of the recently developed binary descriptors [11][12], the barrier of computational cost being prohibitive in real-time applications was overcome.

The experiments involving feature tracking were performed using a pyramid variant of the widely known Kanade-Lucas-Tomasi (KLT) optical flow algorithm first proposed in [14] and extended in [15]. Tracking was initialized using the features detected using the FAST algorithm [10] known for its low computational cost.

The following algorithms were used for feature detection, description and matching: the FAST [10] feature detector paired with the binary BRIEF feature descriptor, the multiscale, L2-norm based SIFT [8] and SURF [9] detectors and descriptors, as well as the ORB multiscale binary detector and descriptor [12].

3 Experiments

The experiments were performed using the data from the ICL-NUIM dataset [16]. The dataset consists of video sequences from a synthetic environment with perfect ground-truth poses of the camera. During the rendering process special care has been given to simulate the artifacts usually present in the images registered by a camera. The synthetic data was used due to the required precision of the ground truth, as even small errors in the reference camera trajectory could corrupt the evaluation of the features localization.

The 'Living Room 0' sequence consisting of 1510 frames was used in the study. The most prominent 200 point features were detected on each of the first 1460 frames. Afterwards the detected features were localized on the following 50 frames.

In the case of the feature tracking approach, the points of interest were detected using the FAST detector. The KLT tracker was used to estimate the positions of the features on the consecutive images. Only the features that were successfully tracked on the i-th frame were analyzed on the $i+1$-th image.

In the case of the feature matching, the features were detected and described using one of the following algorithms: the FAST-BRIEF combination, the ORB, the SURF and the SIFT. The descriptors of the features found on the i-th frame were independently matched against the descriptors of the features detected on each of the following frames, up to the $i+50$-th frame. The match was considered to be successful if the ratio of the distances between the second-best match and the best match was smaller than 0.8.

The experiments were performed on a computer with the Intel i5 processor (2.6 GHz) and 12GB RAM. The resolution of the analyzed images was 640×480.

The precision of the point features' localization on the analyzed frame was evaluated in the terms of the symmetric reprojection error. Consider $\begin{bmatrix} u_0 & v_0 \end{bmatrix}^T$ and $\begin{bmatrix} u_i & v_i \end{bmatrix}^T$ to be the position of the feature on the initial frame and the feature's estimated position on the i-th frame correspondingly. If R_i and t_i stand for the reference rotation and translation between the considered poses of the camera and M is the camera matrix. The fundamental matrix describing the epipolar geometry can be calculated as:

$$F = \left(M^T\right)^{-1} R_i [t_i]_x M^{-1} \tag{1}$$

where $[t_i]_x$ is the matrix representation of a cross product with the vector t_i. The parameters of the epipolar lines on both images can be calculated as:

$$\begin{bmatrix} a_0 & b_0 & c_0 \end{bmatrix}^T = F \begin{bmatrix} u_i & v_i & 1 \end{bmatrix}^T \tag{2}$$

$$\begin{bmatrix} a_i & b_i & c_i \end{bmatrix}^T = \begin{bmatrix} u_0 & v_0 & 1 \end{bmatrix} F \tag{3}$$

Fig. 1. Matching with FAST-BRIEF combination (top) and tracking (bottom). Matches with reprojection error less than 2 are marked white.

Finally, the symmetric reprojection error of the analyzed feature's localization is defined as:

$$d = max\left(\frac{|a_0 u_i + b_0 v_i + 1|}{\sqrt{a_0^2 + b_0^2}}, \frac{|a_i u_0 + b_i v_0 + 1|}{\sqrt{a_i^2 + b_i^2}}\right) \tag{4}$$

Figure 2 presents the comparison of the analyzed approaches in various aspects of feature localization. Subplot (a) shows the average ratio of the features that were successfully localized on the consecutive frames. It is clearly visible that following the tracking approach results in the biggest number of maintained features. It is caused mainly by the fact that the tracking is performed on the frame-to-frame base and an exhaustive search of the features' neighbourhood is performed. In the case of the descriptor matching the most features are maintained by the FAST-BRIEF combination. The biggest number of rejected matches is observed when using the SIFT algorithm.

Subplot (c) shows the ratio of successfully localized features for which the reprojection error is smaller than 1 pixel. It is visible that using the KLT or SIFT gives the best results. The combination of the FAST and the BRIEF algorithms performs slightly worse, followed by the SURF and the ORB. The accuracy of different methods converges as the frames distance increases. If the reprojection error threshold is set to 2 the characteristics of all the methods but the SURF become similar.

It is worth noting that if the threshold is set to 1, the average ratio of correctly localized features exceeds 0.5 for the frame distance of over 20 frames. The same ratio is maintained for over 30 frames if the error threshold is set to 2. This means

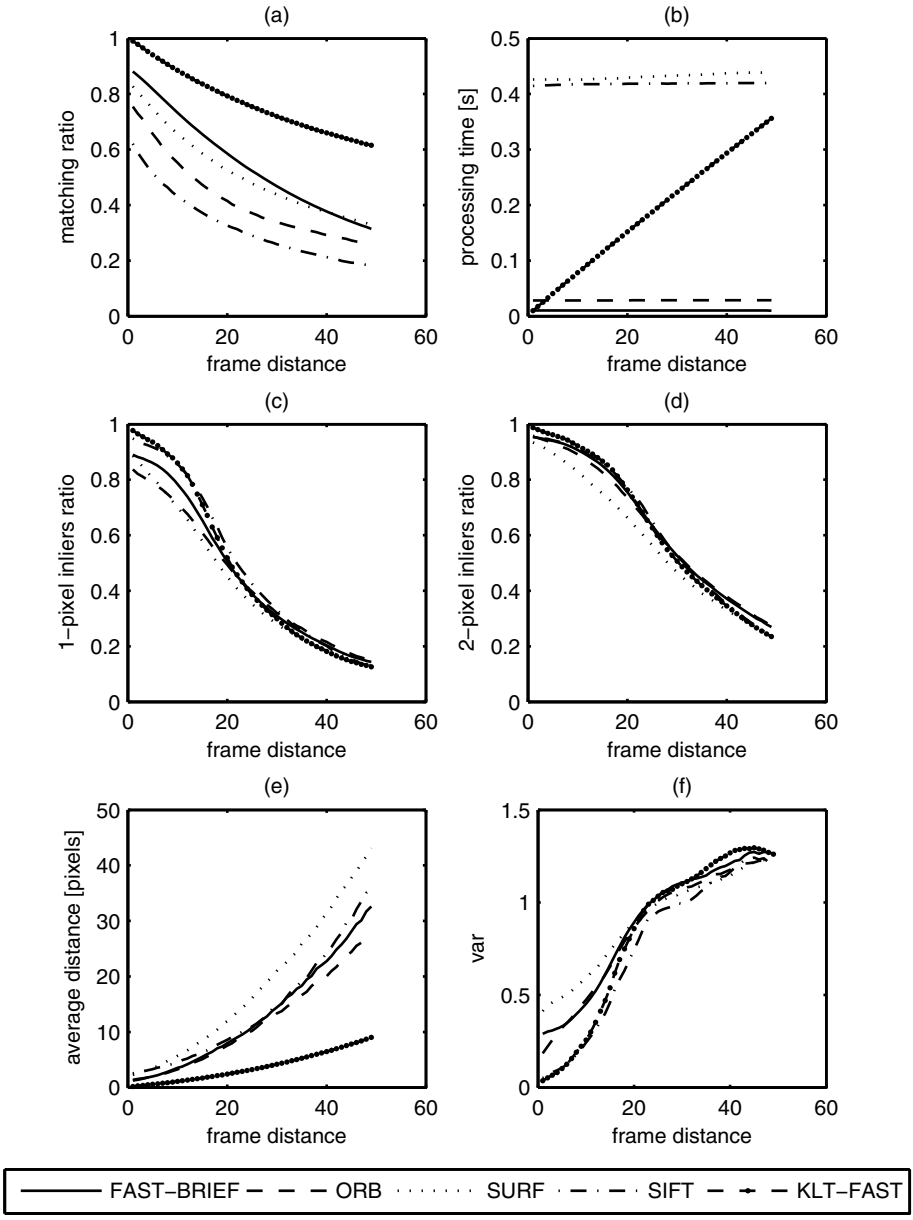

Fig. 2. The comparison of tracking and matching approaches

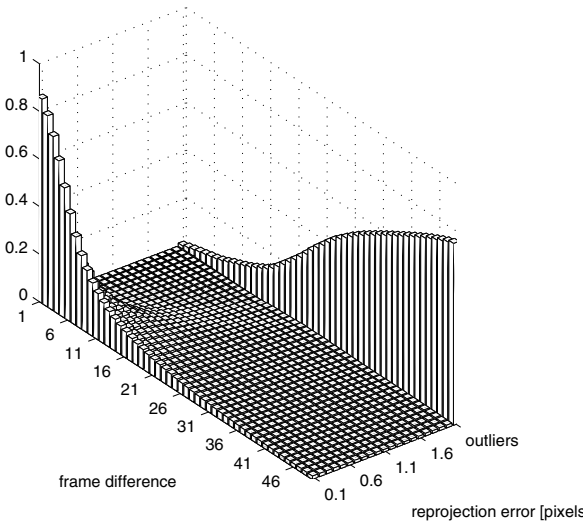

Fig. 3. Normalized histograms of the reprojection error for the KLT

that all the methods can be used for estimation of the camera displacements within a robust estimation framework and they do not differ significantly in the quality of the features' localization.

The biggest difference between the analyzed methods lies in the processing time as shown on subplot (b). In the case of the tracking approach the processing time increases proportionally to the number of analyzed frames. It is caused by the fact that the features are localized on every incoming frame. In the case of the matching the processing time is approximately constant as it comprises of the detection and description of features on only two images and their matching.

It is clearly visible that the matching using the FAST-BRIEF combination outperforms all the other approaches. The KLT is faster than the ORB if less than every fourth frame is analyzed. The SURF and SIFT require over 0.4[s] to match features across two frames. Such long processing time renders the usefulness of those two algorithms in a real-time system doubtful.

The experiments also allowed the estimation of the features' localization uncertainty. Figure 3 presents the concatenated, normalized histograms of the features' reprojection error. The values of the error were dividded into 20 regularly spaced bins between 0 and 2 pixels and the 'outliers' bin. Feature correspondences with the error bigger than 2 were considered outliers. The number of the outliers increases with the frame distance. This also explains the increasing average reprojection error observed in subplot (e) of Figure 2.

Currently, most of the visual navigation systems use robust estimation frameworks (e.g. RANSAC) to find the camera movement hypothesis supported by the

biggest number of inliers. Therefore, only the uncertainty of those inliers needs to be parametrized. It may also be assumed that the features localization is not biased towards any direction. The shape of the normalized histograms suggests that the uncertainty of the features localization on the image can be modelled as an isotropic, additive 2D Gaussian noise. The distribution is considered to be zero-mean and defined only by the diagonal covariance matrix:

$$C = \begin{bmatrix} c & 0 \\ 0 & c \end{bmatrix} \tag{5}$$

Traditionally, the values of c are set to 1. However, they depend on the frame difference and can be estimated from the histograms. Subplot (f) of Figure 2 presents the values of the parameter c for all the considered algorithms. It is visible that the variance of the noise is the smallest if either the KLT or the SIFT was used. The larger variance observed in the case of the other methods is probably caused by the spatial interpolation (SURF) and non-maximal suppression (FAST and ORB).

4 Conclusions

This paper presents the comparison of the feature tracking and matching approaches in the context of visual navigation. The performed experiments clearly show that all the considered methods offer similar accuracy of the features localization. Surprisingly, despite the claimed robustness of the ORB algorithm w.r.t. the in-plane rotation and scale changes, it performed worse than the FAST-BRIEF combination. This is probably caused by the interpolation of features localization across different scales in the ORB detector.

Due to insignificant differences in the accuracy, the selection of the specific algorithm should be based on other criteria. If the processing time is crucial, which is the case in most visual navigation systems, the combination of the FAST detector and BRIEF descriptor is an obvious choice.

The BRIEF and ORB algorithms outperformed the tracking mainly due to the fact that only two frames are analyzed. However, the frame-to-frame analysis can be advantageous. The tracking can be stopped if the number of correctly localized features drops below an assumed threshold like in [3]. This is especially important in the presence of motion blur or rapid changes of the scene which can lead to an abrupt decrease in the number of correctly localized features.

Moreover, the analysis of the features' localization uncertainty was performed. The obtained results will be used in the visual SLAM system to parametrize the observations of the point features.

The future work will focus on two tasks. Firstly, the influence of the number of localized features on the processing time and localization accuracy will be assessed. Secondly, the performance of different feature detectors with the KLT will be evaluated. The obtained results will be used to select the optimal approach for point features localization in the visual SLAM system.

Acknowledgments. This research was financed by the Polish National Science Centre grant funded according to the decision DEC-2013/09/B/ST7/01583, which is gratefully acknowledged.

References

1. Scaramuzza, D., Fraundorfer, F.: Visual odometry: Part I the first 30 years and fundamentals. IEEE Robotics and Automation Magazine **18**(4) (2011)
2. Fraundorfer, F., Scaramuzza, D.: Visual odometry: Part II: Matching, robustness, optimization, and applications. IEEE Robotics & Automation Magazine **19**(2), 78–90 (2012)
3. Nowicki, M., Skrzypczynski, P.: Combining photometric and depth data for lightweight and robust visual odometry. In: 2013 European Conference on Mobile Robots (ECMR), pp. 125–130. IEEE (2013)
4. Strasdat, H., Montiel, J., Davison, A.J.: Scale drift-aware large scale monocular slam. Robotics: Science and Systems **2** (2010)
5. Zhang, Z., Huang, Y., Li, C., Kang, Y.: Monocular vision simultaneous localization and mapping using surf. In: 7th World Congress on Intelligent Control and Automation, WCICA 2008, pp. 1651–1656 (June 2008)
6. Schmidt, A., Kraft, M., Fularz, M., Domagała, Z.: Visual simultaneous localization and mapping with direct orientation change measurements. In: Gruca, A., Czachórski, T., Kozielski, S. (eds.) Man-Machine Interactions 3. AISC, vol. 242, pp. 127–134. Springer, Heidelberg (2014)
7. Lee, J., Cui, X., Lee, S., Kim, H., Kim, H.: Inha: Localization of mobile robots based on feature matching with a single camera. In: 2013 IEEE International Conference on Systems, Man, and Cybernetics (SMC), pp. 2765–2770 (October 2013)
8. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision **60**(2), 91–110 (2004)
9. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: Speeded-up robust features (surf). Computer Vision and Image Understanding **110**(3), 346–359 (2008)
10. Rosten, E., Drummond, T.W.: Machine learning for high-speed corner detection. In: Bischof, H., Leonardis, A., Pinz, A. (eds.) ECCV 2006, Part I. LNCS, vol. 3951, pp. 430–443. Springer, Heidelberg (2006)
11. Calonder, M., Lepetit, V., Ozuysal, M., Trzcinski, T., Strecha, C., Fua, P.: Brief: Computing a local binary descriptor very fast. IEEE Transactions on Pattern Analysis and Machine Intelligence **34**(7), 1281–1298 (2012)
12. Rublee, E., Rabaud, V., Konolige, K., Bradski, G.: Orb: an efficient alternative to sift or surf. In: 2011 IEEE International Conference on Computer Vision (ICCV), pp. 2564–2571. IEEE (2011)
13. Schmidt, A., Kraft, M., Fularz, M., Domagala, Z.: Comparative assessment of point feature detectors and descriptors in the context of robot navigation. Journal of Automation, Mobile Robotics & Intelligent Systems **7**(1) (2013)
14. Shi, J., Tomasi, C.: Good features to track. In: 1994 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 1994), pp. 593–600 (1994)
15. Bouguet, J.Y.: Pyramidal implementation of the lucas kanade feature tracker description of the algorithm (2000)
16. Handa, A., Whelan, T., McDonald, J., Davison, A.: A benchmark for RGB-D visual odometry, 3D reconstruction and SLAM. In: IEEE Intl. Conf. on Robotics and Automation, ICRA, Hong Kong, China (to appear, May 2014)