A Setup for Evaluating Detectors and Descriptors for Visual Tracking

Steffen Gauglitz*
Dept. of Computer Science
University of California,
Santa Barbara

Tobias Höllerer*
Dept. of Computer Science
University of California,
Santa Barbara

Petra Krahwinkler†
Institute of Man-Machine
Interaction
RWTH Aachen University

Jürgen Roßmann†
Institute of Man-Machine
Interaction
RWTH Aachen University

ABSTRACT

In many cases, visual tracking is based on detecting, describing, and then matching local features. A variety of algorithms for these steps have been proposed and used in tracking systems, leading to an increased need for independent comparisons. However, existing evaluations are geared towards object recognition and image retrieval, and their results have limited validity for real-time visual tracking. We present a setup for evaluation of detectors and descriptors which is geared towards visual tracking in terms of testbed, candidate algorithms and performance criteria. Most notably, our testbed consists of video streams with several thousand frames naturally affected by noise and motion blur.

1 INTRODUCTION

Visual tracking is a core component for robot navigation as well as augmented reality. Although it may be accomplished by other means (e.g., template- or model-based), it is in many cases (e.g. [2]) based on detecting, describing, and then matching local features. Many algorithms for these subtasks have been proposed and used in tracking.

However, existing evaluations [4, 5, 6] focus on object recognition and image retrieval rather than tracking: They use low-noise, high-resolution still images, and large databases to match features against. Execution time, a crucial criterion for designing real-time systems, receives little to no attention, and considered algorithms tend to be computationally expensive and often intractable for real-time use; hence their results have limited validity for real-time visual tracking. Most notably, we are not aware of any work that compares the respective algorithms on video streams, which is the setup of interest for visual tracking applications in general and augmented reality in particular.

Our evaluation is oriented towards visual tracking in all of the factors mentioned above: the evaluated algorithms have been proven in real-time applications, the performance measures are chosen with respect to visual tracking, and our testbed consists of video streams affected by noise and motion.

2 EVALUATION SETUP

Ground Truth. To evaluate algorithms on images taken with a moving camera, ground truth information is needed, specifying which point x_j in frame j corresponds to point x_i in frame i. For general 3D scenes, this is very difficult to obtain without a 3D model of the scene. Like most existing evaluations [4, 6] we therefore use planar scenes. where x_i and x_j are related by a homography H_{ij}. However, the methods used by existing evaluations to solve for H_{ij} are not feasible for arbitrary camera motion and/or several thousand frames (the same is true for the setup of [5], which is not restricted to planar scenes).

*e-mail: {sgauglitz,holl}@cs.ucsb.edu
†e-mail: {krahwinkler,rossmann}@mmi.rwth-aachen.de

IEEE International Symposium on Mixed and Augmented Reality 2009
Science and Technology Proceedings
19 -22 October, Orlando, Florida, USA
978-1-4244-5389-4/09/$25.00 ©2009 IEEE

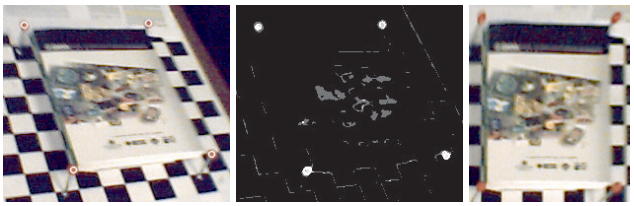


Figure 1: Using the result of an adaptive color model (center), the input image (left) is warped into a canonical reference frame (right).

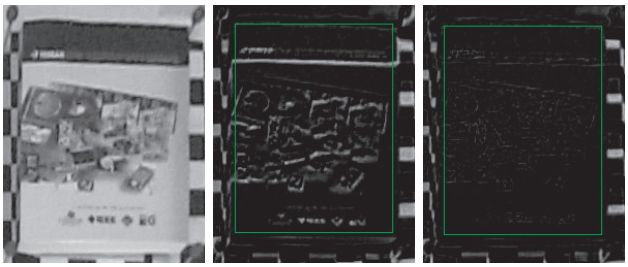


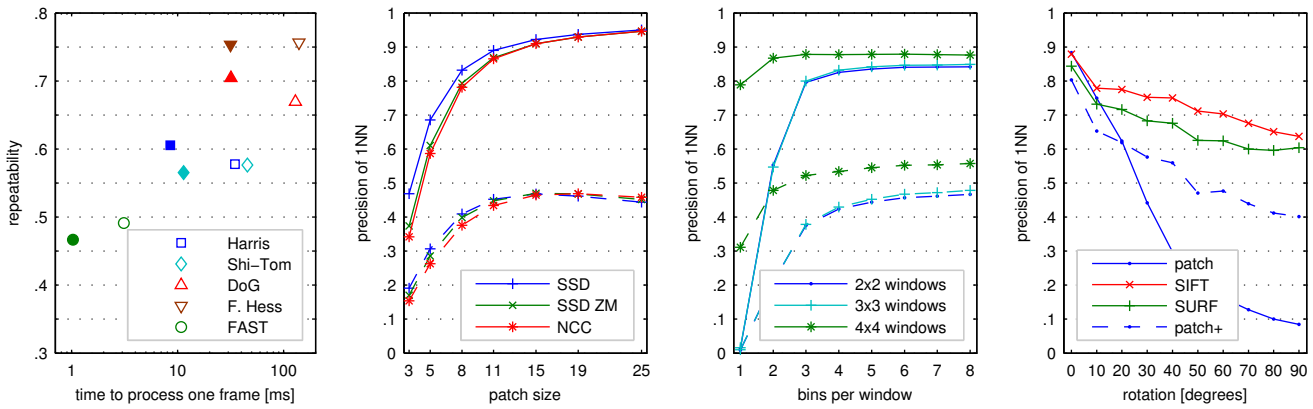
Figure 2: Warped frame (left), difference to reference image before and after image alignment (center and right, respectively). The alignment was substantially improved. For the evaluation, only the area inside the green rectangle is used, not including the markers (which violate the assumption of an unprepared environment) and the surrounding area (for which the homographic warp is incorrect).

For our work, we designed a semi-automatic algorithm to compute H_{ij}: We use four small red balls as markers (chosen because they do not change appearance even for extreme changes in viewpoint and placed such that their centers lie in the plane of the texture) and manually indicate their position in the first frame. They are then tracked using an adaptive color model and template matching (cf. Fig. 1). H_{ij} is computed from the new positions of the four balls, and finally, the warp is refined using image alignment (cf. Fig. 2).

Testbed. The testbed consists of 30 different video streams with a total of 2711 frames, showing five different planar textures in six different motion patterns each (translational movement, in-plane rotation, out-of-plane rotation, zoom, motion blur, as well as unconstrained camera motion exhibiting all of the above). The textures were chosen to encompass features with different levels of contrast and repetitiveness.

The algorithms' performance was measured between consecutive frames, simulating continuous tracking during smooth motion, as well as between randomly chosen frames of a sequence, simulating tracking recovery after failure or re-visiting a previously mapped scene. Including this randomized order, all algorithms and algorithm combinations were evaluated for approximately 30,000 frame pairs.

Performance measures. Given the task of real-time tracking, we consider execution time a crucial performance measure for both detectors and descriptors, especially as faster execution and thus



(a) detectors: repeatability vs. time (b) patch, different configurations (c) SIFT, different configurations (d) precision vs. in-plane rotation

Figure 3: Exemplary results. (a) Repeatability vs. processing time for 640x480 (empty markers) and 320x240 pixels (filled), evaluated for consecutive frames. Repeatability between random frames is significantly lower, and the ranking of the detectors is different. (b) Precision of matching with image patch, varying the patch's size and similarity measure (SSD sum of squared distances, ZM SSD zero-mean SSD, NCC normalized cross correlation). Here, solid lines indicate the average performance on consecutive frames of smooth motion (i.e. with very small baseline distances), while dashed lines indicate the average performance on a set of random frame pairs with potentially wide baseline distances. (c) Precision of matching with SIFT. Solid/dashed lines as in (b). (d) Precision in the case of in-plane rotation. "patch+" is image patch with SURF's orientation assignment.

higher framerates might in return decrease prediction uncertainty and thus improve tracking. Although timings are platform- and implementation-specific, differences of orders of magnitude (such as shown in Fig. 3a) are unlikely to change unless special hardware acceleration is used.

The most relevant quality criterion for detectors is repeatability [6]. For descriptors, ROC curves (precision/recall) are commonly used in the object recognition community [4, 5]. However, due to speed constraints, virtually all visual tracking systems evaluate only the first nearest neighbor (1NN), hence its precision is the most relevant criterion in this context. Additionally, we measure the percentage of successful tracking for any of the possible detector-descriptor combinations.

Candidate algorithms. The feature detectors that we evaluated so far are: Harris Corner Detector, Shi-Tomasi Corner Detector, FAST, Difference of Gaussians (DoG) and Fast Hessian; the descriptors are: image patch (with and without scale information and orientation assignment), SIFT and SURF. As the implementation is crucial (especially when measuring time), we used the original implementations where available, otherwise we used publicly available and widely used versions.

3 Evaluations

Algorithm configurations. To ensure that all algorithms are optimally configured for the task of tracking, we first evaluated a variety of parameters. For detectors, comprehensive results of this evaluation may be found in a technical report [1]. For descriptors, two exemplary evaluations are shown in Figs. 3b and 3c. For example, Fig. 3c shows that the dimensionality of SIFT can be greatly reduced (compared to the default configuration) with hardly influencing performance. This has been exploited e.g. by [7], although our results suggest to use a different configuration (4x4 windows with 2 or 3 bins instead of their choice of 3x3 windows with 4 bins).

Comparison. We then compared the detectors' and descriptors' performances in all scenarios that our testbed offers. We evaluated both stages individually as well as in combination. Due to space constraints and as we plan on extending the evaluations (cf. Section 4), we present only one example of the results: Fig. 3d shows the average 1NN precision as a function of the in-plane rotation between the two frames. In this case, SIFT and SURF clearly outperform a simple image patch description, even if the latter uses the same (here: SURF's) orientation assignment. Our evaluations also show that the alleged robustness of the histogram-based descriptors does not help in the case of out-of-plane rotation, where a patch descriptor performs slightly better, while the performance of all descriptors decreases quickly.

4 Ongoing and Future Work

We are extending our evaluations to more algorithms. In particular, we are planning to evaluate the classification-based approaches around [3], which share the paradigm of local features, but "recognize" them using previously trained classifiers rather than computing and matching a descriptor vector. It would also be interesting to extend the evaluations to template-based matching, though in that case, it would have to encompass a much wider scope, as the two paradigms have less in common.

References

[1] S. Gauglitz and T. Höllerer. In-depth evaluation of popular interest point detectors on video streams. Technical Report 2009-08, Department of Computer Science, UC Santa Barbara, May 2009.

[2] G. Klein and D. Murray. Parallel tracking and mapping for small AR workspaces. In *Proc. Sixth IEEE and ACM Intl. Symposium on Mixed and Augmented Reality (ISMAR'07)*, Nara, Japan, November 2007.

[3] V. Lepetit and P. Fua. Keypoint recognition using randomized trees. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 28(9): 1465–1479, 2006.

[4] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, Oct. 2005.

[5] P. Moreels and P. Perona. Evaluation of features detectors and descriptors based on 3D objects. *Intl. Journal of Computer Vision*, 73(3):263–284, 2007.

[6] C. Schmid, R. Mohr, and C. Bauckhage. Evaluation of interest point detectors. *Intl. Journal of Computer Vision*, 37(2):151–172, 2000.

[7] D. Wagner, G. Reitmayr, A. Mulloni, T. Drummond, and D. Schmalstieg. Pose tracking from natural features on mobile phones. In *Proc. 7th IEEE and ACM Intl. Symposium on Mixed and Augmented Reality (ISMAR'08)*, Cambridge, UK, Sept. 15–18 2008.