AnySURF: Flexible Local Features Computation

Eran Sadeh-Or and Gal A. Kaminka
Computer Science Department
Bar Ilan University, Israel

The use of computer vision in autonomous robotics has been studied for decades. Recently, applications such as autonomous vision-based vehicle navigation [1], 3-D localization and mapping [6,4,3] and object recognition [5] have gained popularity due to the combination of increased processing power, new algorithms with real-time performance and the advancements in high quality, low-cost digital cameras. These factors enable autonomous robots to perform complex, real-time, tasks using visual sensors.

Such applications are often based on a local feature matching algorithm, finding point correspondences between two images. There are many different algorithms for feature matching, however in recent years there is a growing research on algorithms that use local invariant features (for a survey see [10,8]). These features are usually invariant to image scale and rotation and also robust to changes in illumination, noise and minor changes in viewpoint. In addition, these features are distinctive and easy to match against a large database of local features.

Unfortunately, existing algorithms for local feature matching [2,6,7] are designed under the assumption that they will run to completion and only then return a complete result. Many of these algorithms therefore require significant computational resources to run in real-time. As we show in the experiments, this prohibits some of the algorithms from being used in current robotic platforms (where computation is limited). For instance, a Nao[1] humanoid robot computing the full set of features in an image of size 640×480 requires 2.4 seconds using a state-of-the-art implementation of the SURF algorithm [2,9].

Note, however, that for many robotics applications, even partial results—a subset of all features in the image—would have been sufficient (for example, to estimate the pose of the robot for obstacle detection). On the other hand, being able to invest computation time in getting higher-quality results is also important, e.g., in object recognition or in building accurate maps. Indeed, robots can benefit from computationally-flexible algorithms, where the computation time is traded for the accuracy requirements of the task. To do this, simply interrupting the algorithm when needed is not enough: We need to guarantee that the results of the algorithm would necessarily monotonically increase in quality, given additional computation time. This class of algorithms is called *Anytime* [11].

In this paper we present AnySURF, an anytime feature-matching algorithm, which can accumulate results iteratively, with monotonically increasing quality and minimal overhead. We achieve flexibility by re-designing several major steps

[1] http://www.aldebaran-robotics.com

in the SURF algorithm [2], mainly the feature search process and the order of interest point detection. We additionally discuss the design choices underlying AnySURF.

We evaluate the use of AnySURF in a series of experiments. We first demonstrate that non-anytime feature matching indeed suffers from significant computation time on limited platforms (including, in particular, the Nao humanoid robot). Then, we contrast different design choices for AnySURF, and analyze its performance profile under different image types. We also demonstrate the usability of AnySURF in computing approximate homography.

References

1. DARPA grand challenge. http://www.darpa.mil/grandchallenge/index.asp, 2007.
2. H. Bay, T. Tuytelaars, and L. V. Gool. SURF: speeded up robust features. In *Computer Vision – ECCV 2006*, pages 404–417. 2006.
3. D. Chekhlov, M. Pupilli, W. Mayol-cuevas, and A. Calway. Real-time and robust monocular SLAM using predictive multi-resolution descriptors. *In 2nd International Symposium on Visual Computing*, 2006.
4. A. Davison, I. Reid, N. Molton, and O. Stasse. MonoSLAM: Real-Time single camera SLAM. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(6):1052–1067, 2007.
5. D. G. Lowe. Object recognition from local Scale-Invariant features. In *Proceedings of the International Conference on Computer Vision - Volume 2*, page 1150. IEEE Computer Society, 1999.
6. D. G. Lowe. Distinctive image features from Scale-Invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, Nov. 2004.
7. K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(10):1615–1630, 2005.
8. K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. V. Gool. A comparison of affine region detectors. *Int. J. Comput. Vision*, 65(1-2):43–72, 2005.
9. A. Orlinski. Pan-o-matic - automatic control point creator for hugin.
10. T. Tuytelaars and K. Mikolajczyk. Local invariant feature detectors: a survey. *Found. Trends. Comput. Graph. Vis.*, 3(3):177–280, 2008.
11. S. Zilberstein. Using anytime algorithms in intelligent systems. *AI Magazine*, 17(3):73–83, 1996.