Comparison of detectors and distance metrics for pose estimation

Comparación de detectores y métricas de distancia para la estimación de pose

MARTÍNEZ-DÍAZ, Saúl†*

Tecnológico Nacional de México/Instituto Tecnológico de La Paz, División de estudios de Posgrado e Investigación

ID 1st Author: *Saúl Martínez-Díaz* / **ORC ID:** 0000-0003-4962-5995, **Researcher ID Thomson**: Q-7112-2019, **CVU CONACYT ID**: 175255

DOI: 10.35429/JSI.2022.19.6.11.16 Received September 14, 2022; Accepted December 29, 2022

Abstract

In many artificial vision applications, it is necessary to know the pose (rotation and translation) of the camera with respect to some object in the real world. To know this pose, many algorithms are based on the detection and matching of common points of interest in two or more images. For that reason, it is necessary to have adequate algorithms for point detection and a robust metric for pose estimation. This paper presents a comparative analysis of three of the most popular algorithms for point detection and two popular metrics. In the detectors, the robustness to geometric distortions, robustness to noise and processing speed were compared. In the metrics robustness to noise and processing speed were compared.

Point detectors, Pose estimation, Artificial vision

Resumen

En muchas aplicaciones de visión artificial es necesario conocer la pose (rotación y traslación) de la cámara con respecto a algún objeto del mundo real. Para conocer dicha pose, muchos algoritmos se basan en la detección y emparejamiento de puntos de interés, comunes en dos o más imágenes. Por esa razón, es necesario contar con algoritmos adecuados para la detección de puntos y una métrica robusta para la estimación de la pose. En este artículo se presenta un análisis comparativo de tres de los algoritmos más populares para la detección de puntos y dos métricas populares. En los detectores se compararon la robustez a distorsiones geométricas, robustez a ruido y velocidad de procesamiento. Para las métricas se compararon la robustez al ruido y la velocidad de procesamiento.

Detectores de puntos, Estimación de pose, Visión artificial

Citation: MARTÍNEZ-DÍAZ, Saúl. Comparison of detectors and distance metrics for pose estimation. Journal of Systematic Innovation. 2022. 6-19: 11-16

* Correspondence to Author (e-mail: saul.md@lapaz.tecnm.mx)

† Researcher contributing as first author.

Introduction

In many applications, such as autonomous robot navigation (Wang, Liu, & Li, 2015) (Knudson & Tumer, 2011), simultaneous localization and mapping (SLAM) (Mur Artal & Tardos, 2017) (Mur Artal & Tardos, Visual-inertial monocular slam with map reuse, 2017) and augmented reality (AR) (Chatzopoulos, Bermejo, Huang, & Hui, 2017), pose estimation is an important topic.

In this context, pose represents the position and orientation of a three-dimensional object with respect to a reference system, in real world. Usually, orientation is represented by a 3x3 rotation matrix and position is represented by a 3x1 translation vector.

Most techniques of pose estimation rely on detection of key points. The basic steps for pose estimation, based on key points, using two images from a video sequence are:

– Detect key points in the images

– Match corresponding points in both images performing a nearest neighbor search

– Apply a robust estimator, such as Random Sample Consensus (RANSAC) algorithm (Torr & Zisserman, 2000), to reduce mismatches

– Compute pose using information of matched points, camera parameters and geometric techniques.

Key points must be visually significant points that can be easily identified in each image. An ideal key point detector should find salient image regions, despite change of viewpoint. Each detected point is represented by a vector of features, called descriptor, which is extracted from such point and its neighbors.

A good descriptor should make it possible to uniquely distinguish each point. Corners can be used for this porpoise; however, other better algorithms have been introduced, being some of the most important:

Scale-Invariant Feature Transform (SIFT) (Lowe, 1999), Features from Accelerated Segment Test (FAST) (Rosten & Drummond, 2005), Speeded-Up Robust Features (SURF) (Bay, Ess, Tuytelaars, & Van Gool, 2008), Binary Robust Invariant Scalable Keypoints (BRISK) (Leutenegger, Chli, & Siegwart, 2011), Binary Robust Independent Elementary Features (BRIEF) (Calonder, Lepetit, Strecha, & Fua, 2010) and Oriented FAST and rotated BRIEF (ORB) (Rublee, Rabaud, Konolige, & Bradski, 2011). Some detectors include its own descriptors.

The desirable characteristics of detectors and descriptors are speed, robustness to geometrical distortions and robustness to noise.

Note that selection of a good detector and descriptor are very important tasks for pose estimation. This paper presents a comparison between the main detectors and descriptors used for estimation of pose. Besides, a comparison between two distance metrics, namely Sum of Squared Difference (SSD) and Sum of Absolute Difference (SAD), is included.

Basic Concepts

In this section we present a brief review of the algorithms to be compared, including detectors FAST, SURF and BRSK, and the two metrics, SSD and SAD.

1. Features from Accelerated Segment Test (FAST)

FAST use a circle with a perimeter of 16 pixels, around the corner candidate pixel p. Pixels are numbered clockwise, starting from top center pixel. The algorithm classifies p as a corner if there is a set of n contiguous pixels in the circle, all within a range $p \pm t$, where t is a threshold value (normally, $n = 12$ is chosen). To speed up execution, four pixels are examined at the positions 1, 5, 9 and 13. For p to be a corner, at least three of these values must be in the indicated range, Otherwise, p is discarded. The number of features detected, and the speed of detection is determined by the threshold t. In addition, the 16-pixel circle can be used as a feature vector.

2. Speeded-Up Robust Features (SURF)

SURF searches for points of interest by applying a Gaussian filter to image and calculating the determinant of its Hessian matrix H. To achieve scale invariance, Gaussian filters of different sizes are applied, divided into octaves. The maximum values obtained from the determinant of H are the indicators of the location of the points of interest. Once the points of interest have been obtained, to achieve rotation invariance, the direction of the gradient at each point is calculated using Haar wavelets.

The SURF descriptor computes orientation using the Haar wavelets in the *x* and *y* directions, in a circular region of radius 6*s*, where *s* is the scale of the point of interest.

Individual responses are weighted with a Gaussian function centered on the point of interest. The dominant orientation is obtained as the sum of all responses within a radius of $\pi/3$, using a sliding window. The vertical and horizontal responses are added and the vector with the greater value is kept. Descriptor is constructed by forming a square region of size 20*s* around the point of interest, using the dominant orientation and a Gaussian weighting. Then, the region is divided into 4x4 subregions and, within each subregion, the Haar response of points spaced 5x5 in both directions is calculated. Next, in each subregion, the vertical and horizontal responses and their absolute values are added. Finally, the vector is formed by these four components (sums) of the 4x4 regions, giving a total of 64 elements.

3. Binary Robust Invariant Scalable Keypoints (BRISK)

BRISK is method for key point detection, description, and matching. To reduce computational cost, like SURF, points of interest are identified at different scales divided into octaves, using a saliency criterion. The location and the scale of each key point are obtained using a quadratic function fitting. For description, a sampling pattern is applied at the neighborhood of each key point. The pattern consists of points lying on appropriately scaled concentric circles. Orientation is determined processing local intensity gradients. Finally, the oriented BRISK pattern is used to assemble the binary BRISK descriptor.

4. Sum of Squared Difference and Sum of Absolute Difference

Matching process requires to compare each point in the first image with all points of the second image. To determine which point matches other, a nearest neighborhood search is performed. If the distance of the closest point is less than a certain threshold, the points are matched. Besides, an algorithm to reduce outliers is necessary to reduce mismatches.

To compute distance between two points, the most used metrics are Sum of Squared Difference (SSD) and Sum of Absolute Difference (SAD). Let (x,y,z) and (x’,y’,z’) two points to be compared, SSD and SAD are defined respectively as

$$\mathrm{SSD} = (x - x')^2 + (y - y')^2 + (z - z')^2 \qquad (1)$$

$$\mathrm{SSD} = |x - x'| + |y - y'| + |z - z'| \qquad (2)$$

Methodology

1. Selection of detector

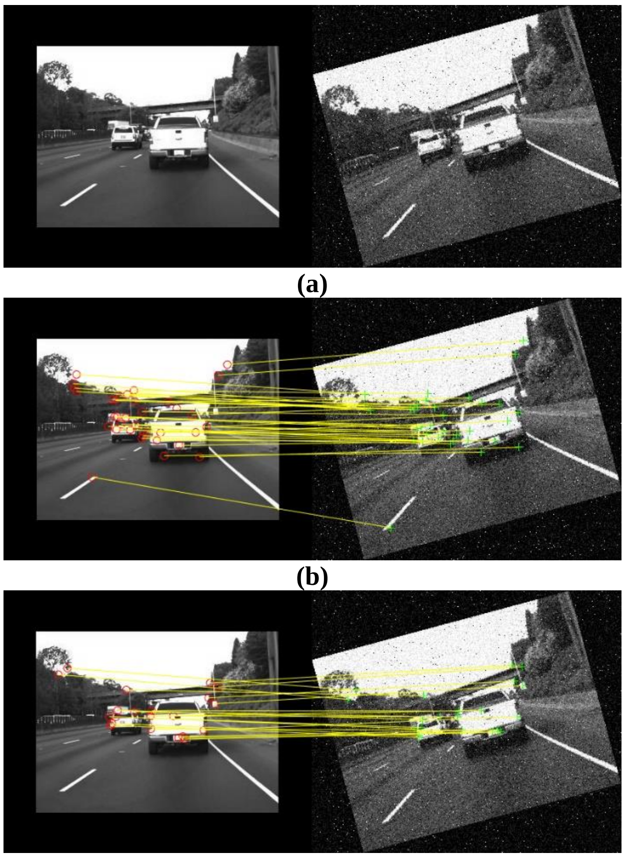
Robustness of keyframe-based method greatly depends on techniques used for detection and matching. To choose the best detector, we tested three popular algorithms: Features from Accelerated Segment Test (FAST), Speeded-Up Robust Features (SURF) and Binary Robust Invariant Scalable Keypoints (BRISK).

We tested speed and robustness of algorithms with three different images: cameraman, Lena, and highway. First, to verify robustness to geometric distortions, we change 10% the scale of a reference image and rotate it from -15 to 15 in steps of 5 degrees; next, to probe robustness to noise, we add zero-mean Gaussian additive noise with variance of 0.01 and impulsive noise with prob-ability of 0.05 to the rotated and scaled image; then, with the three proposed detectors, we searched and matched interest points in both, the reference and modified images; finally, we computed error due to mismatches and we measured execution time. To obtain statistically correct results, 30 trials of each experiment for different realizations of random noise were performed and results were averaged. All experiments were performed in a CORE i7 processor at 2.6 GHz.

2. Selection of matching algorithm

With the extracted features it is necessary to match points in at least two images. For this, the matching algorithm performs a nearest neighbor search by computing the pairwise distance between feature vectors. Then, it selects the strongest matches respect to a stablished threshold and returns its indices. If the number of outliers is low, the number of iterations and time consumption of RANSAC estimation method can be reduced. For this reason, we first compare robustness of two popular distance metrics, namely Sum of Squared Difference (SSD) and Sum of Absolute Difference (SAD). To test images with real rotations and noise, we match SURF features obtained from 600 stereo pairs images of sequence 00, 01, 02 and 03 from KITTI dataset (Geiger, Lenz, Stiller, & Urtasun, 2013).

We applied Random Sample Consensus (RANSAC) algorithm between each pair of images with both metrics. RANSAC is a general and very successful robust estimator used for this purpose. Since the algorithm uses a random process, to obtain statistically correct results, we perform 30 statistical trials with each pair of images and average results.



(a)

(b)

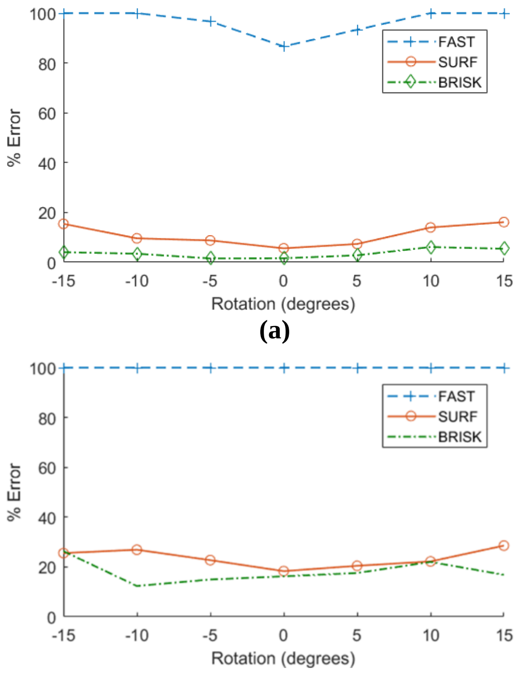
(c)

Figure 1 Example of matched points with a) FAST, b) SURF and c) BRISK detectors for highway image

Results

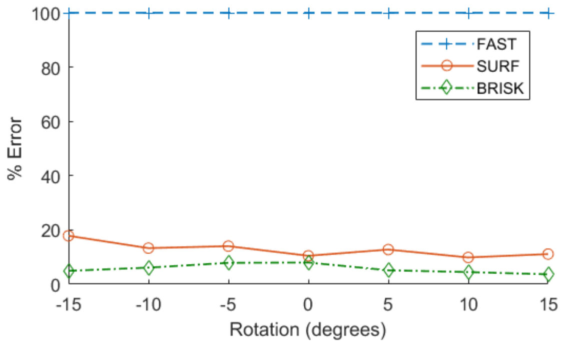
In this section, we show results of tested algorithms. For detectors comparison, figure 1 is an example of matched images. As can be seen in figure 1(a), due to noise and geometric distortions, FAST detector was unable to correctly match any point in these images. On the other hand, as can be appreciated in figures 1(b) and 1(c), SURF and BRISK correctly detect many points.

Figure 2 shows percentage of matching error of the three detectors with respect to rotation angle for the three tested images: a) cameraman, b) Lena and c) highway. Size of images were 256x256, 512x 512 and 240x320 pixels, respectively. Table 1 contains results of total execution time (in seconds) with each detector. Because of noise, FAST detector is unable to correctly match most points. BRISK detector reaches the best results in terms of error for al-most all tests; however, time consumption is very high compared with the other detectors (almost ten times, in some cases). On the other hand, SURF detector offers a good tradeoff between performance and time; therefore, we selected this detector for the distance metric comparison.



(a)

(b)



(c)

Figure 2 Percentage of mismatches of FAST, SURF and BRISK detectors with a) Cameraman, b) Lena and c) highway images

	Cameraman	Lena	Highway
FAST	7.82	11.59	8.1
SURF	5.47	17.69	6.12
BRISK	57.7	82.47	59.16

Table 1 Comparation of time (in seconds) of FAST, SURF and BRISK detectors

Method	SSD	SAD
Iterations seq. 00	48.24	34.96
Iterations seq. 01	32.32	22.25
Iterations seq. 02	39.81	28.82
Iterations seq. 03	39.36	31.44
Average iterations	39.9325	29.3675
Total time (sec)	1065.37	848.49

Table 2 Comparation of iterations and processing time of SSD and SAD

For matching algorithm comparison, table 2 shows the results of number of iterations in each sequence, average number of iterations and total time consumption for the two matching algorithms. As can be seen, SAD required fewer average iterations and less time consumption, therefore is a good candidate for matching metric in the keyframe-based pose estimation.

Conclusions

In this paper, we show a comparison of three popular detectors and two matching metrics. Selection of a good detector is crucial to get a reliable pose estimation. The detectors compared were FAST, BRISK and SURF. Although BRISK has a lower error rate, SURF offers a good tradeoff between performance and processing time. The matching metrics compared were SSD and SAD. In our experiments, since SAD is a more robust metric, it required fewer average iterations and, therefore, less time consumption than SSD.

Acknowledgment

This work has been funded by Tecnológico Nacional de México [grant number 11432.21-P]. Texto redactado en Times New Roman No.12, espacio sencillo.

References

Bay, H., Ess, A., Tuytelaars, T., & Van Gool, L. (2008). SURF:Speeded Up Robust Features. *Computer Vision and Image Understanding*, 346–359.

Calonder, M., Lepetit, V., Strecha, C., & Fua, P. (2010). BRIEF: Binary Robust Independent Elementary Features. *European Conference on Computer Vision* (págs. 778–792). Springer.

Chatzopoulos, D., Bermejo, C., Huang, Z., & Hui, P. (2017). Mobile augmented reality survey: from where we are to where we go. *IEEE Access*, 6917-6950.

Geiger, A., Lenz, P., Stiller, C., & Urtasun, R. (2013). Vision meets robotics: The KITTI dataset. *International Journal of Robotics*, 553–572.

Knudson, M., & Tumer, K. (2011). Adaptive navigation for autonomous robots. *Robotics and Autonomous Systems*, 410-420.

Leutenegger, S., Chli, M., & Siegwart, R. (2011). BRISK: Binary Robust Invariant Scalable Keypoints. *IEEE International Conference ICCV*. IEEE.

Lowe, D. G. (1999). Object Recognition from Local Scale-Invariant Features. *International Conference on Computer Vision* (págs. 1150–1157). Kerkyra: IEEE.

Mur Artal, R., & Tardos, J. D. (2017). ORB-SLAM2: an open-source SLAM system for monocular, ste-reo and RGB-D cameras. *IEEE Transactions on Robotics*, 1255-1262.

Mur Artal, R., & Tardos, J. D. (2017). Visual-inertial monocular slam with map reuse. *IEEE Robotics and Automation Letters*, 796-803.

Rosten, E., & Drummond, T. (2005). Fusing Points and Lines for High Performance Tracking. *IEEE International Conference on Computer Vision* (págs. 1508–1511). IEEE.

Rublee, E., Rabaud, V., Konolige, K., & Bradski, G. (2011). ORB: An Efficient Alternative to SIFT or SURF. *International Conference on Computer Vision* (págs. 2564–2571). Barcelona: IEEE.

Torr, P. H., & Zisserman, A. (2000). MLESAC: A new robust estimator with application to estimating Image geometry. *Computer Vision and Image Understanding*, 138–156.

Wang, K., Liu, Y., & Li, L. (2015). Vision-based tracking control of underactuated water surface robots without direct position measurement. *IEEE Transactions on Control Systems Technology*, 2391-2399.