# Evaluation of Feature Detection Algorithms for Structure from Motion

Natasha Govender
Mobile Intelligent Autonomous Systems
CSIR
Pretoria
Email: ngovender@csir.co.za

***Abstract*—Structure from motion is a widely-used technique in computer vision to perform 3D reconstruction. The 3D structure is recovered by analysing the motion of an object, based on its features, over time. The typical steps involved in SFM are feature detection, feature matching and determining the motion and pose of the cameras. For each step, a number of different algorithms may be used.**

**Little research has however been done into the effectiveness of the different feature detection algorithms such as Harris corner detectors and feature descriptors such as SIFT (Scale Invariant Feature Transform) and SURF (Speeded Up Robust Features) given a set of input images. This paper implements state-of-the-art feature detection algorithms and evaluates their results on a given set of input images. The evaluation will be preformed by comparing the calibration data, the fundamental matrix and the rotation and translation errors extracted from each algorithm with ground truth data.**

## I. Introduction

Structure from motion (SFM) is used to recover the 3D structure and camera motion from a set of moving images. SFM techniques are used in applications ranging from photogrammetric survey [1] to the automatic reconstruction of virtual reality models from video sequences [2] and for the determination of camera motion for use in augmented reality.

SFM algorithms involves a number of different steps which include feature detection, feature matching and estimation of the camera motion and pose. For each step various algorithms may be used. Feature detection is a low-level process which is most often the starting point for computer vision applications. Thus, the success of an algorithm depends substantially on this initial step. Ideally for SFM, feature detectors need to be robust and be able to locate the same features in successive images irrespective of image rotation, scaling or changes in illumination. The features detected are then matched across images. This allows for the estimation of the calibration matrix of the camera, the fundamental matrix and to predict the camera motion. Therefore the accuracy and robustness of the feature detector and matching algorithm, have a direct impact on the accuracy of these estimations.

The algorithms implemented in this paper are Harris corner detectors, the Kanade-Lucas-Tomasi (KLT) feature tracker, SIFT and SURF. Evaluations have been conducted using different feature detection and descriptor algorithms but no evaluations have been done on how the performance of these algorithms impact on SFM algorithms. A standard correlation matching algorithm was used. Input images were captured using a single Prosilica camera mounted on an autonomous vehicle which was then driven around the CSIR campus. The images were dewarped to remove radial distortion and then used in the experiments.

The layout of the paper is as follows; The next section looks at the background literature for feature detectors. In Section III we discuss the different feature detection and descriptor algorithms implemented for the experiments. Section IV describes the experiments conducted. In Section V we discuss the results obtained and conclude in Section VI.

## II. Background

A comprehensive overview of the current methods for the extraction of features can be found in [3]. Schmid [4] accomplished a practical comparison of feature detectors using the original implementations of the authors. The operators of Forstner [5], Cottier [6], as well as Harris [7] were evaluated quantitatively. It was found that the Harris operator was the most stable of all. Hall [8] formalized a definition of saliency under scale changes and evaluated the Harris, Lindeberg [9] and Harris-Laplacian corner detectors as well.

Lowe [10] described image feature generation with SIFT. Mikolajczyk [11] compared SIFT descriptors, steerable filters, differential and moment invariants, complex filters and cross-correlation for different types of interest points. He observed that SIFT descriptors perform best and steerable filters come second. Zuliani [12] proposed a unifying description and mathematical comparison of the Harris, Noble, Kanade-Lucas-Tomasi(KLT) and Kenney point detectors.

In [13] the performance of feature detectors and descriptors for images of 3D objects viewed under different viewpoints, lighting and scaling conditions were evaluated on a large database. The detectors evaluated were the Harris detector, Hessian detector and difference of Gaussian filters. The descriptors used were SIFT, PCA_SIFT, steerable filters and shape context descriptors. They found the best overall choice was using an affine-rectified detector followed by SIFT or a shape-context descriptor. These detectors and descriptors were the best when tested for robustness to changes in viewpoint, change in lighting and change in scale. However, the selection of an optimal procedure remains difficult, since the results substantially depend on the respective implementation.

## III. Feature Detectors and Descriptors

A feature is used to denote a piece of information which is relevant for solving the computational task related to a certain application. It can refer to specific structures in the image itself, ranging from simple structures such as points or edges to more complex structures such as objects. Some detection algorithms such as SIFT also use a feature descriptor which is used to uniquely describes a feature in an image. This aids in the matching process as it helps to reduce ambiguities.

### A. Harris Corner Detector

Corner detection was originally proposed by [14]. The algorithm tests each pixel in the image to see if a corner is present, by considering how similar a patch centered on the pixel is to nearby, largely overlapping patches. The similarity is measured by taking the sum of squared differences (SSD) between the two patches. A lower number indicates more similarity. The corner strength is defined as the smallest SSD between the patch and its neighbors (horizontal, vertical and on the two diagonals). If this number is local maximum , then a feature of interest is present. As pointed out by Moravec, one of the main problems with this operator is that it is not isotropic i.e. if an edge is present that is not in the direction of the neighbours, then it will not be detected as an interest point. Harris and Stephens [7] improved upon Moravec's corner detection algorithm by considering the differential of the corner score with respect to direction directly instead of using shifted patches.

### B. SIFT

SIFT was first described by [10]. It is used to detect and describe features in images. The SIFT features are local and based on the appearance of the object at particular interest points and are invariant to image scale and rotation. They are also robust to changes in illumination, noise and minor changes in viewpoint. The steps involved in SIFT are:

- **Scale-space extrema detection**: Interest or keypoints are detected in this step.
- **Keypoint Localization**: Scale-space extrema detection produces too many keypoint candidates, some of which are unstable. A detailed fit to the nearby data for accurate location, scale, and ratio of principal curvatures is done. This information allows points to be rejected that have low contrast (and are therefore sensitive to noise) or are poorly localized along an edge.
- **Orientation Assignment**: Each keypoint is assigned one or more orientations based on local image gradient directions. This is the key step in achieving invariance to rotation as the keypoint descriptor can be represented relative to this orientation and therefore achieve invariance to image rotation.
- **Keypoint Descriptor**: This step creates a histogram of local oriented gradients around the interest point and stores the bins in a 128-dimensional vector to compute highly distinctive descriptors for the keypoints.

### C. SURF

SURF is a scale and rotation invariant interest point detector and descriptor [15]. It is inspired by the SIFT descriptor. The standard version of SURF is faster than SIFT and claimed by its authors to be more robust against different image transformations than SIFT.

First, interest points are selected at distinctive locations in the image, such as blobs and T-junctions. Next, the neighbourhood of every interest point is represented by a feature vector. This descriptor has to be distinctive and, at the same time, robust to noise, detection errors, and geometric and photometric deformations. Finally, the descriptor vectors are matched between different images. The matching is often based on a distance between the vectors, e.g. the Mahalanobis or Euclidean distance.

The descriptor describes a distribution of Haar-wavelet responses within the interest point neighbourhood. Integral images are used here for speed gains and only 64 dimensions are used, reducing the time for feature computation and matching, and increasing simultaneously the robustness.

### D. Kanade-Lucas-Tomasi

KLT feature tracker, which is sometimes referred to as the Kanade-Tomasi corner detector, is based on the early work of Lucas and Kanade [16] and was later developed fully by Tomasi and Kanade [17]. Here good features are located by examining the minimum eigenvalue of each 2 by 2 gradient matrix, and features are tracked using a Newton-Raphson method of minimizing the difference between the two windows. Multiresolution tracking allows for relatively large displacements between images. The corner detector is strongly based on the Harris corner detector. The authors show that for image patches undergoing affine transformations, the minimum of the two magnitudes of eigenvalues is a better measure of corner strength than the function suggested by Harris.

## IV. Experiments

The implemented feature detection algorithms were applied to a set of input images (1152x864 resolution) obtained from a Prosilica camera mounted on an autonomous vehicle which was driven through a portion of the CSIR. Figure 1 is an example of an input image used in our experiments. The images obtained were dewarped to remove radial distortion around the edges. This step requires that the camera be accurately calibrated. The calibration matrix contains the intrinsic parameters of the camera such as the focal length, image format and principal point.

A standard correlation matching algorithm was then applied. MLESAC [18] was the used to remove outliers and find the maximum likelihood to estimate the fundamental matrix. This was then followed by a non-linear refinement step. The fundamental matrix is a 3x3 matrix which expresses the relationship between any two images of the same scene that restricts where the projection of points from the scene can occur in both images. Using the fundamental matrices

Fig. 1. An example of an input image

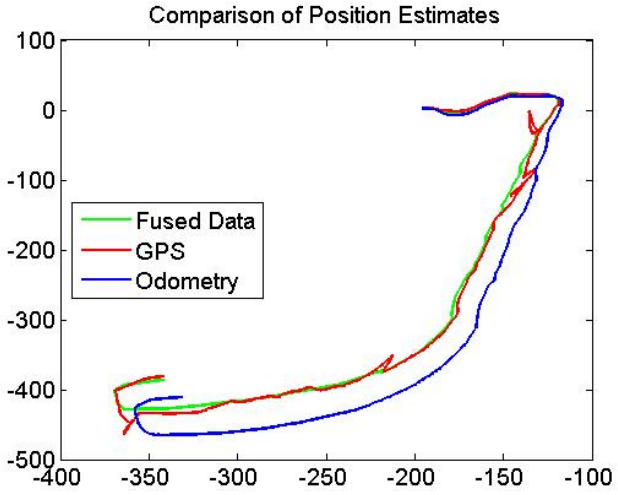


Fig. 2. The vehicle odometry fused with the GPS data to produce the ground truth odometry

extracted from each algorithm we calculated the reprojection error generated by each feature detector. The reprojection error is the geometric error related to the image distance between a projected point (using the data extracted from the algorithms) and a measured one (ground truth data).

The autonomous vehicle used to capture the images has built-in odometry. To ensure the accuracy of this data, a GPS was also placed on the vehicle. This GPS data was then fused with the odometry from the vehicle to obtain a fairly accurate plot of the path taken by the vehicle. The rotation and translation vectors were extracted from this data and used as rotation and translation ground truth. These were then compared against the results extracted from each algorithm.

Figure 2 displays the ground truth odometry of the path taken by the autonomous vehicle.

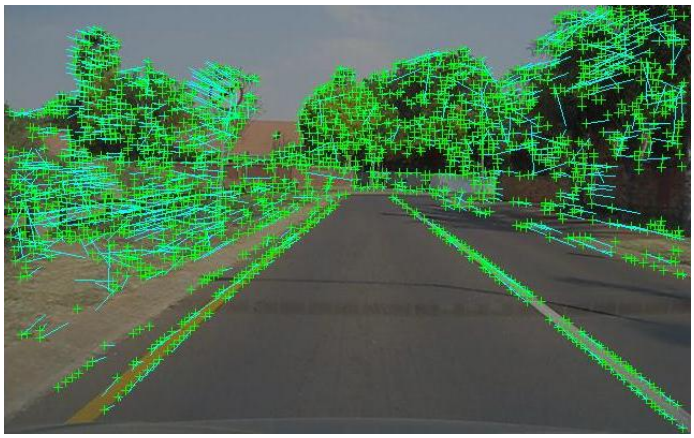

Fig. 3. An input image with detected features and matches

TABLE I
AVERAGE TRANSLATION ERROR

| Feature Detector | Avg. Translation Error X | Avg. Translation Error Y |
|---|---|---|
| Harris Corner Detector | 0.848 | 0.1942 |
| KLT Tracker | 0.3985 | 0.04835 |
| SIFT | 0.3786 | 0.4073 |
| SURF | 0.7787 | 0.545 |

## V. RESULTS

Each algorithm implemented was run on the same set of input images. Figure 3 depicts an input image with the features detected indicated in green and those which have been matched to features in a previous image are highlighted in light blue.

### A. Experiment 1

The rotation and translation vectors for each image were extracted. The euclidean distance was taken between the ground truth translation and rotation data and those extracted from each algorithm after all the data was normalised. The results were then averaged over the number of input images. The results for the translation errors are displayed in Table I and in Table II for rotational errors.

SIFT and the KLT tracker have the lowest errors for both translation and rotation. This indicates that for this set of inputs, these two detectors are more accurate and robust in detecting features in each image than SURF and Harris corner detectors i.e they were able to track the camera motion more accurately.

### B. Experiment 2

The fundamental matrix estimated by each algorithm was also inspected. Traditionally the accuracy of the fundamental

TABLE II
AVERAGE ROTATION ERROR

| Feature Detector | Avg.Rotation Error X | Avg.Rotation Error Y |
|---|---|---|
| Harris Corner Detector | 0.815 | 0.9854 |
| KLT Tracker | 0.8192 | 0.0119 |
| SIFT | 0.5949 | 0.2230 |
| SURF | 1.5257 | 1.2398 |

TABLE III
REPROJECTION ERROR FOR EACH DETECTOR

| Feature Detector | Reprojection Error |
|---|---|
| Harris Corner Detector | 7.89 |
| KLT Tracker | 7.21 |
| SIFT | 6.923 |
| SURF | 9.134 |

TABLE IV
AVERAGE EPIPOLE DISTANCE

| Feature Detector | Epipole Distance |
|---|---|
| Harris Corner Detector | 10.3232 |
| KLT Tracker | 9.6872 |
| SIFT | 9.0372 |
| SURF | 15.895 |

matrix was assessed by checking how well the parameters fit the observed data, but as pointed out in [19] this is the wrong criterion as the aim is to find the set of parameters that best fit the unknown true data. The parameters of the fundamental matrix themselves are not of primary importance, rather it is the structure of the corresponding epipolar geometry. Therefore it makes little sense to compare two solutions by directly comparing the difference in their fundamental matrices. Thus, we compare the differences in the associated epipolar geometry weighted by the density of the given matching points. This is done using the estimated fundamental matrix to calculate the reprojection error. The results are displayed in Table III.

SIFT has the lowest reprojection error. Hence, the fundamental matrix estimated by SIFT is closest to the ground truth.

We also used the fundamental matrix to calculate the average distance in pixels from the true point in each image to that yielded by the estimate of the fundamental matrix.

Here the Harris corner detector, KLT and SIFT all perform relatively well. SURF descriptor has the worst performance in both experiments involving the fundamental matrix indicating that the fundamental matrix estimated by SURF is far from the ground truth.

## VI. CONCLUSION

The set of input images used in these experiments were captured in an outdoor environment during the day with changes in illumination and speed of the vehicle. Using this specific dataset we have shown that SIFT performs better than Harris corner detectors, SURF and the KLT tracker. It's translation and rotation errors were the lowest, indicating that it was able to successfully locate the same features in consecutive images and thus fairly accurately estimating the camera motion. The fundamental matrix estimated by the SIFT algorithm was also closest to the ground truth.

## REFERENCES


[1] K.Kraus, *Photogrammetry*, Dummler, Ed., 1997, vol. I and II.

[2] R.I.Hartley and A.Zisserman, *Multiple View Geometry in Computer Vision, 2nd Ed*. Cambridge University Press, 2004.

[3] C.Schmid, R.Mohr, and C.Bauckhage, "Comparing and evaluating interest points," in *ICCV*, 1998, pp. 230–235.

[4] C. Schmid, R.Mohr, and C.Bauckhage, "Evaluation of interest point detectors," in *Int. Journal of Computer Vision*, vol. 37, no. 2, 2000, pp. 151–172.

[5] W. Forstner, "A framework for low level feature extraction," in *Third Eurpoean conference on Computer Vision*, 1994, pp. 383–394.

[6] J.C.Cottier, "Extraction et appariements robustes des points d'interet de deux images non etalonnees," LIFIA-IMAG-INRIA,Rhone-Alpes, Tech. Rep., 1994.

[7] C.Harris and M.J.Stephens, "A combined corner and edge detectors," in *Alvey Vision Conference*, 1988, pp. 147–152.

[8] D.Hall, B.Leibe, and B. Schiele, "Saliency of interest points under scale changes," in *British Machine Vision Conference*, 2002, pp. 646–655.

[9] T.Lindeberg, "Feature detection with automatic scale selection," in *International Journal of Computer Vision*, vol. 30, no. 2, 1998, pp. 79–116.

[10] D. Lowe, "Distinctive image features from scale invariant keypoints," in *International Journal of Computer Vision*, ser. 91-110, vol. 60, no. 2, 2004.

[11] K.Mikolajczyk and C.Schmid, "A performance evaluation of local descriptors," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2003, pp. 253–263.

[12] M.Zuliani, C.Kennedy, and B.Manjunath, "A mathematical comparison of point detectors," in *Conference on Computer Vision and Pattern Recognition Workshop*, vol. 11, 2004, pp. 172–178.

[13] P.Moreels and P.Perona, "Evaluation of feature detectors and descriptors based on 3d objects," in *ICCV*, 2005.

[14] H. Moravec, "Obstacle avoidance and navigation in the real worl by a seeing robot rover," Carnegie-Mellon University,Robotic Institue, Tech. Rep., 1980.

[15] H.Bay, T.Tuytelaars, and L.V.Gool, "Surf: Speeded up robust features," in *Proceedings of the ninth Conference on Computer Vision*, May 2006.

[16] B.D.Lucas and T. Kanade, "An iterative registration technique with an application to stereo vision," in *International Joint Conference on Artificial Intelligence*, April 1981.

[17] C.Tomasi and T.Kanade, "Detection and tracking of point fetaures," Carnegie Mellon, Tech. Rep., April 1991.

[18] P. Torr and A.Zisserman, "Mlesac: A new robust estimator with application to estimating image geometry," in *Computer Vision and Image Understanding*, 2000, pp. 138–156.

[19] P.H.S.Torr, "A structure and motion toolkit in matlab: interactive adventures in s and m," Microsoft Research, Tech. Rep., 2004.