In 16th Portuguese Conference on Pattern Recognition (RecPad 2010)
Vila Real, Portugal, October 2010

Random features *vs* Harris Corners in Real-Time Visual Egomotion Estimation

Nuno Moutinho, Ricardo Ferreira, Alexandre Bernardino and José Gaspar
Instituto de Sistemas e Robótica, Instituto Superior Técnico, 1049-001 Lisboa, Portugal
{nmoutinho, ricardo, alex, jag}@isr.ist.utl.pt

Abstract

We compare Randomly Selected (RanSel) features with Harris Corners within a visual egomotion estimation framework. Harris corners have been extensivelly used in visual egomotion estimation systems due to a good tracking stability. However, to compute these features the whole image has to be processed. Instead, we propose the use of randomly selected points which are virtually costless to obtain. Despite tracking individual RanSel features is not as stable as Harris corners, we show that, when integrated in a time-filtering scheme, they provide similar results at a much faster rate. We have performed experiments using a synthetic setup with ground-truth and discuss the advantages of using RanSel features.

1. Introduction

Feature tracking is a central component of Visual Egomotion Estimation. There are many tracking methodologies, such as the well known Lucas-Kanade Tracker [1], with good performances using different types of features (Harris corners [4], SURF [2] or SIFT features [5]).

Different types of features have different computation costs. Systems with real-time demands must select features that simultaneously provide good precision and fit in the available computational resources. Due to its quality and speed, the Harris corner detector is widely used in visual motion estimation methods such as Visual SLAM (Simultaneous Localization and Mapping) [3]. However, the demand for implementations in low cost and low power embedded devices requires more economic algorithms.

In this paper we question whether it is preferable to detect N complex (time consuming) features which can be reliably tracked or $M > N$ simpler features which are not so reliably tracked but still, when integrated in an adequate processing pipeline, achieve similar performance. We propose taking this idea to the extreme by using Randomly Selected (RanSel) image features which require virtually no time to compute. We compare the advantages and disadvantages of using such features in a Visual Odometry setup instead of using the Harris algorithm.

2. Egomotion Estimation

Our system is based on an Extended Kalman Filter (EKF) which estimates the linear V_r and angular W_r velocities of the robot's head (egomotion) as well as the $3D$ position of all the currently observed features Y_n, collected in a state vector $X_r = [V_r\ W_r\ Y_1\ ...\ Y_N]$. It can be described in four main steps: (i) Feature Detection, (ii) EKF Prediction, (iii) Feature Tracking and (iv) EKF Update.

Feature detection, step (i), occurs only when the egomotion estimation system is initialized or some features are lost. Old features still visible and the new detected ones are fed into the EKF. The EKF Prediction, step (ii), provides predictions for the locations of the features in the next image frame, assuming a constant velocity model. The Feature Tracking, step (iii), uses these predictions as the center of a neighborhood where to search for the features and obtain the measurements. The EKF Update, step (iv), uses these measurements as input to update the system's state X_r.

In this paper we focus mainly in step (i), Feature Detection. Two types of features were considered, Harris corners and RanSel. RanSel features are randomly selected points from a Uniform Distribution supported on the whole image, to ensure an adequate scattering. In both cases, the detected features are characterized by descriptors composed by all pixels in a squared neighborhood.

Feature descriptors carry the necessary information for the Feature Tracking step (iii), described by:

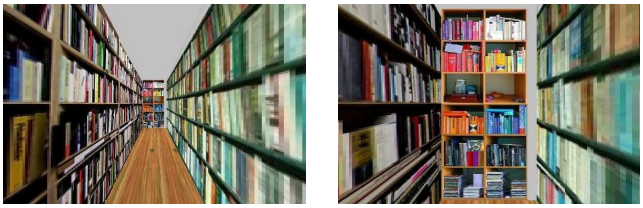
$$(\Delta u, \Delta v)^* = \arg \max_{\Delta u, \Delta v} d\{w(I_1; u, v), w(I_2; u+\Delta u, v+\Delta v)\}$$

where I_1, I_2 are two images, $w(I; u, v)$ is a window centered at (u, v) and $d(A, B)$ represents the normalized cross-correlation between image *patches* A and B. A *matching* is detected if the maximized response d is greater than a certain threshold λ.

In regions with low texture, RanSel features may result in weak descriptors, which lead the tracker to exhibits a "sliding effect" due to the similarity between the neighbor textures. However, if a sufficient number of points is selected, the likelihood of acquiring only weak features is reduced. Since generating N random feature locations is always faster than computing N Harris Corners, the RanSel algorithm is expected to be more time efficient and, within the same time interval, allow tracking a larger number of points. In the following section, we perform experiments that will illustrate these ideas.

3. Results

Our tests use a synthetic ground-truth stereo sequence (10 seconds, $30fps$) representing a corridor (Fig. 1).



(a) Left camera, frame 0 (b) Left camera, frame 300

Figure 1: The setup used to obtain the stereo sequence.

In each implementation we chose the greatest number of features allowing a processing time below $0.033s$ per frame. This led to a number of 19 Harris corners and 27 RanSel points. The remaining algorithm paraments (EKF, $19\times19\,pixel$ descriptor window, etc.) were the same in both cases so we can isolate the influence of the feature selection mechanism. Figure 2 shows, for the whole sequence, the estimated components of the linear velocity against the ground truth. The generated motion was absent of angular velocities.

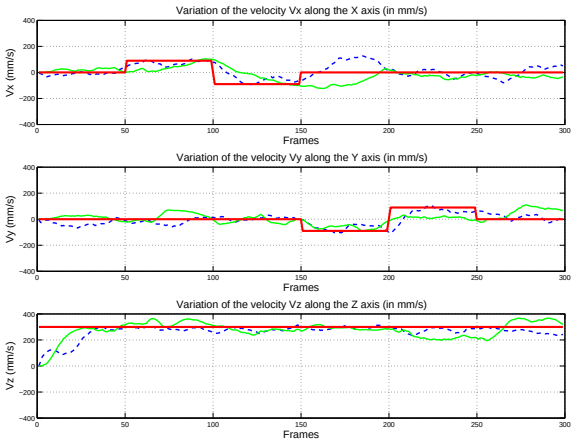


Figure 2: Estimation of the linear velocity components V_x, V_y and V_z, using 19 Harris corner features (in green) and 27 RanSel features (in blue). The real values are in red.

In the Harris case, most of the features were located in the middle of the image, corresponding to a distant plane (see figure 1), where the Harris Corner Detector responses were stronger. Notice that the filter took about 50 frames to converge to the real values, when responding to a *step* and that some perturbations occured during the estimation process (V_x plot, in green). The features detected in the center of the image were the ones with the higher depth uncertainty and this explains the perturbations that appear from frame 200 to 300 in the V_z estimation. The location of the features, allied with their low number, contributed in a negative way to the slow convergence of the filter and to the appearance of some estimation perturbations.

In the RanSel case more features were found within the $33ms$ time constraint, so a constant number of 27 features were used. Unlike the Harris case, the filter took only 20 frames to converge to the real values (plots in blue). Perturbations, although still existing, were of lesser magnitude. The use of more features together with their random selection prevents that many of them fall in regions of low texture. This contributs positively to the final estimation.

The mean absolute errors of the estimations are represented in the table 1.

Features	$e_{v_x}(mm/s)$	$e_{v_y}(mm/s)$	$e_{v_z}(mm/s)$
Harris	44.9681	39.5190	47.7623
RanSel	35.4729	29.5843	35.5216

Table 1: Mean absolute errors of the linear velocity components.

The estimations obtained by the system using RanSel features have a lower error, for each linear velocity component. Thus it is preferable to have more features, even if some are of lower quality, rather than have lesser but of greater quality. The time consumed in the search for image feature corners could be used to acquire more features, scattered throughout the image, which gives more accurate results.

4. Conclusions

In this paper we showed that it is possible to obtain better results using Randomly Selected features instead of Harris Corner features, simply by trading feature accuracy for a larger number of points. This approach is faster, simpler and is presented as an alternative to nowadays algorithms used in real time systems.

Acknowledgments

This work was partially funded by FCT (ISR/IST pluriannual funding) through the PIDDAC Program funds.

References

[1] Simon Baker and Iain Matthews. Lucas-kanade 20 years on: A unifying framework. *International Journal of Computer Vision*.

[2] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. *CVIU-Computer Vision and Image Understanding*, 110(3):346–359, 2008.

[3] Andrew Davison, Ian Reid, Nicholas Molton, and Olivier Stasse. Monoslam: Real-time single camera slam. *IEEE*, 29:16, 2007.

[4] Chris Harris and Mike Stephens. A combined corner and edge detector. *The Plessey Company*, pages 147–152, 1988.

[5] David G. Lowe. Object recognition from local scale-invariant features. *International Conference on Computer Vision*, 1999.