Monocular methods in stereo visual odometry

Ilan Shimshoni, Ehud Rivlin, Alex Kreimer

October 11, 2015

Abstract

We propose a new algorithm for stereo visual odometry based on monocular methods

1 Introduction

TBD

2 Related Work

TBD

3 Algorithm Description

The algorithm in 3.3.1 solves VO in stereo setting. It relies on the estimation and decomposition of the essential matrix and the invariance of the cross-ratio under projectile transformation so we overview these first in 3.1 and 3.2

3.1 Monocular motion estimation

Let C_1 and C_2 be the two camera frames. R describes the orientation of C_1 in as seen from C_2. q_1 is a direction (line of site of the origin) from C_2 to C_1 described in C_2 (see Figure 1)

Thus for every pair of image correspondences $x \in C_1, x' \in C_2$ holds

$$x'^T K^{-1}[q]_\times R K^{-1} x = 0$$

where K is an intrinsic parameters matrix and $E = [q]_\times R$ is the essential matrix. We estimate the fundamental ($F = K^{-1}[q]_\times R K^{-1}$) using a standard technique such as normalized 8-point algorithm) and then strip it down to essential matrix. Further on we decompose the essential to obtain motion parameters q and R. R may be determined exactly while q only up to scale There are 4 different decompositions, which produce the same essential, but only one is correct. This ambiguity is resolved by imposing the chierality constraint upon the scene points.

An in-depth review of fundamental matrix and its uncertainty estimation is given in [5]

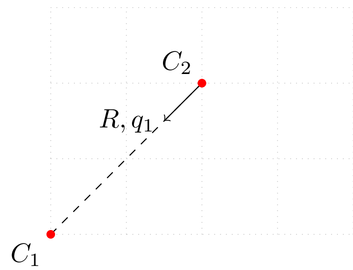

Figure 1: Two views

3.2 Cross-ratio

We propose to exploit the fact that most of the time the car goes more or less straight forward. This special case of camera motion possesses peculiar cross-ratio related properties.

It is a known fact [3] that if camera motion is a pure translation the projections of the world point will "surf" along the epipolar line (towards the epipole or away from it depending on the sign of the translation vector). It was also shown [1] that the cross ratio of feature locations and the vanishing point is exactly the same as the one of the camera centers and the ideal point and thus may be used as an additional constraint in motion estimation.

We face two issues: decide when to use the cross ratio constraint and how to incorporate it into the motion estimation process.

To address the first question we fit a line into the three last feature locations and use the residuals to make a decision. Thus, given a triplet of (homogeneous) feature locations p_i, p_i', p_i'' and the epipole e (see figure 2) we fit a line $w \in R^3$ by solving linear orthogonal regression (using QR decomposition and SVD):

$$w_i^* = \underset{w}{\operatorname{argmin}} \; \|A_i w_i\|_2 \text{ s.t. } w_i^2[1] + w_i^2[2] = 1$$

where $A_i = [p_i, p_i', p_i'', e]^T$. The residuals are given by (the summation is over all the features that are available in last three images):

$$\hat{r}_t = \frac{1}{N} \sum_{i=1}^{N} A_i w_i^* \tag{1}$$

Thus the value of $\hat{r}_t$ may be used to quantitatively assess how "purely translational" is the motion at time t.

The cross ratio for feature point i is given by:

$$Cr(p_i, p_i', p_i'', e) = \frac{\|p_i'' - p_i\| \|e - p_i'\|}{\|p_i' - p_i\| \|e - p_i''\|}$$

Let O, O', O'' be the centers of the cameras for p_i, p_i', p_i'' respectively and let V_∞ be the ideal point of camera motion. Thus for every feature point i holds:

$$Cr(O, O', O'', V_\infty) = Cr(p_i, p_i', p_i'', e)$$

Let us denote $Cr_t = Cr(O, O', O'', V_\infty)$.

Since the locations of the features are noisy we compute average cross-ratio value:

$$\hat{Cr} = \frac{1}{N}\sum_{i=1}^{N} Cr(p_i, p'_i, p''_i, e)$$

And thus for time t we have:

$$Cr_t = \hat{Cr} \tag{2}$$

We will use both 1 and 2 in the motion parameter fitting procedure.

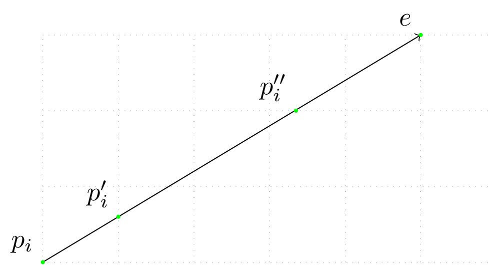

Figure 2: Feature motion feature along the epipolar line during camera translation. The cross ratio $Cr(p_i, p'_i, p''_i, e) = \frac{\|p''_i - p_i\|\|e - p'_i\|}{\|p'_i - p_i\|\|e - p''_i\|}$ has the same value for all features in the image.

3.3 Stereo setup

3.3.1 Stereo motion estimation

Problem statement Let $C_t/C'_t \in \mathbf{SE}(3)$ denote the pose of the left/right camera (respectively) at time t as seen in the world coordinate frame (usually placed at C_1). The rig is moving rigidly, i.e., there is $T_t \in \mathbf{SE}(3)$ s.t. $C_t = T_t * C_{t-1}; C'_t = T_t * C'_{t-1}$ (see Figure 3). Our goal is to estimate T_t given the images taken by the camera at times $\{t, t-1 \ldots t-k\}$

The algorithm presented in 3.1 may determine rotation completely and translation up to scale. In case of a stereo rig we can also determine the scale of the translation. Below is the algorithm outline:

Algorithm 1 (initial estimate)

1. Estimate $T_t = [R_t, q_t]$ using the algorithm in 3.1
2. Estimate $T'_t = [R'_t, q'_t]$ the same way
3. Let c_1, c_2 be the real scales of q_1, q_2. We solve for scale by enforcing $c_1 q_1 + c_2 q_2 = t_0$ (we assume that all vectors are given in the same frame, e.g., the C_1). The same equations may be written in matrix form $Qc = t_0$. Where $Q = [q_1, q_2] \in R^{3\times 2}$ and $c = [c_1, c_2]^T$. Since the system has 3 constraints and 2 variables we solve LS problem instead (using SVD) $c^* = \underset{c}{argmin}\, \|Qc - t_0\|$

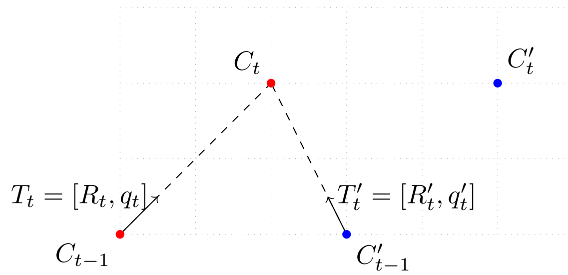

Figure 3: Motion of a stereo rig

3.4 Essential Matrix refinement step

We follow approach proposed in [2] to refine camera motion parameters. Let $r_i(E)$ be the Sampson's error approximation:

$$r_i(E) = \frac{x_i'^T E x_i}{\sqrt{(x_i'^T E)_0^2 + (x_i'^T E)_1^2 + (Ex_i)_0^2 + (Ex_i)_1^2}}$$

If point localization errors are approximately Gaussian, given N correctly matched features $\{(x_i, x_i')|i = 1 \dots N\}$ the MLE of E is approximately the point where $\sum_{i=1}^{N} r_i(E)^2$ is minimized.

It is common to minimize functions in the form of sum of squares using the Levenberg-Marquardt (LM) algorithm [4], a dynamically damped version of Gauss-Newton. It is also common to use robust cost functions to cope with outliers (which break Gaussian residual assumption). This is known as Iteratively Reweighted Least Squares, or IRLS [3]. To minimize a function $\sum_{i=1}^{N} C(r_i)$ for an arbitrary cost function C using IRLS, weights $\{w_i\}$ are chosen so that $(w_i r_i)^2 = C(r_i)$. The function $\sum_{i=1}^{N}(w_i r_i)^2$ is minimized by LM, with weights recomputed at each iteration.

E is parameterized as a function $E(q, t)$ of a unit translation vector t, and a rotation, which is expressed as a quaternion q. Unit quaternions represent 3D rotations as 4D vectors. 4D unit vectors form the differential manifold $\mathbb{S}^3$ (a unit sphere in R^4).

At each iteration, the manifold $\mathbb{S}^3$ is parameterized as three orthogonal vectors τ_1, τ_2, τ_3 tangent to the sphere $\mathbb{S}^3$

We define the following objective:

$$F = \sum_{i=1}^{N_1} r_i(E_1)^2 + \sum_{j=1}^{N_2} r_j(E_2)^2 + \|Qc - t_0\|^2 + \frac{\lambda}{w(\hat{r_t})}(Cr_t - \hat{C}r)^2$$

3.5 Implementation Details

TBD

4 Results

TBD

References

[1] Ronen Basri, Ehud Rivlin, and Ilan Shimshoni. Visual homing: Surfing on the epipoles. *International Journal of Computer Vision*, 33(2):117–137, 1999.

[2] Tom Botterill, Steven Mills, and Richard Green. Refining essential matrix estimates from ransac. In *In Proceedings of Image and Vision Computing New Zealand*, 2011.

[3] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004.

[4] Donald W Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *Journal of the Society for Industrial & Applied Mathematics*, 11(2):431–441, 1963.

[5] Zhengyou Zhang. Determining the epipolar geometry and its uncertainty: A review. *International journal of computer vision*, 27(2):161–195, 1998.