Simultaneous In-Plane Motion Estimation and Point Matching Using Geometric Cues Only

Pierre Fite Georgel
CAMP - TU München *

Adrien Bartoli
Université d'Auvergne †

Nassir Navab
CAMP - TU München

Abstract

We present a novel approach that, given two sets of unmatched keypoints, simultaneously estimates the in-plane camera motion and keypoint matches without using photometric information. Standard approaches estimate the epipolar geometry based on putative matches, first established with photometric information, then accepted or rejected using the epipolar constraint.

Our method discretizes the space of essential matrices at different levels. It searches for the essential matrix and keypoint matches which are the most geometrically coherent. We maximize geometric coherence, that we define as the number of points that can be matched based on the epipolar and unicity constraints. We applied this general framework to sets of images acquired by a moving tripod. We present promising results on simulated and real data.

1. Introduction

A core task in computer vision is to compute the relative camera motion between two views from image features. It has a wide area of application from image stitching to special effects, medical imaging and image categorization. Motion estimation from keypoints has been widely studied [10]. Keypoints are extracted automatically from the images. Putative matches are estimated using correlation directly from image intensity or via descriptors such as SIFT [9] or SURF [1]. State of the art techniques verify the putative matches using RANSAC [5] by estimating a model from a minimal number of points with algorithms such as the 8-point [6], 7-point [18] or the recent 5-point [15]. These robust methods were improved many times for example using oriented geometry [3] or using statistical considerations [11].

Unfortunately, putative matches cannot always be estimated. For example, in the presence of strong perspective changes, massive change of illumination or glare, or when different cameras are combined such as a Time-of-Flight camera with a regular CCD camera. Previous work has been done to overcome the putative matching problem by simultaneously finding a transformation and point matches. For an affine transformation, the RAST method [2] and Geometric Hashing [17] are quite popular. Soft Assign [14] has been successfully applied to affine transformation estimation and deformable warps but also to full camera pose estimation; *e.g.* estimation of the transformation from a 3D / CAD model to an image. The recent blind PnP [12] uses pose priors and error propagation to estimate plausible 2D - 3D matches and the full pose simultaneously. All these approaches have in common the fact that the transformation they try to estimate maps a point to another point, which is not the case in relative motion estimation when points are in relation with epipolar lines. [4] uses Markov Chains Monte-Carlo simulation and EM to maximize a likelihood over structure and motion parameters. This method does not handle clutter. [11] addresses the problem for relative pose estimation from a statistical point of view. [7] proposes a method that discretizes the parameter space. It has global convergence, but requires known point matches and does not deal with wrong matches. To summarize, all previous methods for camera motion estimation through epipolar geometry use photometric information.

We show that point matches and relative motion can be estimated simultaneously without using photometric information. Our framework is based on geometric cues only. It locally finds the 'best' motion despite the ambiguity of the point to epipolar line constraint. We provide a set of essential matrices to this local matching process. We study the in-plane camera motion case, for instance a camera mounted on a tripod where the height, roll and tilt are fixed. This system has two parameters. We use a quad-tree strategy to subdivide the essential matrix space. From a given essential matrix, putative matches are drawn using covariance estimate and guided matching. These matches are then validated with spectral clustering based on a new similarity measure. This is based on an 'essential distance' that we introduce. Finally, we enforce the unicity constraint by maximizing a similarity measure. Using the obtained matches

*e-mail: Pierre.Georgel@Gmail.Com
†e-mail: Adrien.Bartoli@Gmail.Com

we finally estimate the epipolar geometry by minimizing a robust cost.

Paper organization. We first present our method in section 2. Section 3 then gives implementation details and results. We conclude in section 4. In appendix A we describe our 'essential distance' and in supplementary material the proof of the overlapping properties and in a new compact support kernel used in the paper.

Notation. Matrix are in upper case bold (*e.g.* $\mathbf{A}$) and vector in lower case bold (*e.g.* $\mathbf{a}$). We consider two cameras: a source $\mathcal{S}$ and a target $\mathcal{T}$. $\square(\mathbf{a},\mathbf{b}) \subset \mathbb{R}^2$ is the quad formed by the two points $\mathbf{a} \in \mathbb{R}^2$ and $\mathbf{b} \in \mathbb{R}^2$. $\mathbf{o}_{\square(\mathbf{a},\mathbf{b})} \in \mathbb{R}^2$ is the center of the quad $\square(\mathbf{a},\mathbf{b})$ and $\mathbf{c}_{\square(\mathbf{a},\mathbf{b})} \in \mathbb{R}^{2\times 4}$ its four corners. $\mathbb{E}$ is the set of Essential Matrices [8]; it is a variety of dimension N ($N \leq 5$) in $\mathbb{R}^9$ (the set of 3×3 matrices).

2. The Proposed Method

The goal of our method is to estimate an essential matrix $\widetilde{\mathbf{E}} \in \mathbb{E}$ from two sets of unmatched image points ($\{\mathbf{q}_k\} \subset \mathcal{S}$ and $\{\mathbf{p}_l\} \subset \mathcal{T}$). In order to perform this task we sample $\mathbb{E}$ and then estimate putative matches from a given epipolar geometry with parameters $\mathbf{u}$. Samples for $\mathbf{u} \in \mathbb{R}^N$ are obtained through a discretization $\mathcal{E}$ of $\mathbb{E}$. We first describe our camera setup, its parameterization and the developed subdivision scheme. We then explain the matching procedure and finally the robust cost we minimize.

2.1. Discretizing the Motion Model

Camera setup. We consider a camera mounted on a tripod. The camera is oriented such that it is parallel to the horizon. The camera and tripod system remains rigid over time; only the system moves. The camera motion thus has two degrees of freedom: an horizontal translation and a single rotation around the y-axis (see figure 1 for a schematic). This is the usual motion of a tripod. We suppose that the cameras are internally calibrated (*i.e.*, the 3×3 matrices of intrinsics $\mathbf{K}_\mathcal{S}$, $\mathbf{K}_\mathcal{T}$ known). We can thus parametrize the essential matrix with two angles $\mathbf{u} = [\theta, \alpha]^\top$. These define a relative rotation $\mathbf{R}(\mathbf{u})$ and translation $\mathbf{t}(\mathbf{u})$ as

$$\mathbf{R}(\mathbf{u}) = \begin{bmatrix} \cos(\pi-\theta-\alpha) & 0 & \sin(\pi-\theta-\alpha) \\ 0 & 1 & 0 \\ -\sin(\pi-\theta-\alpha) & 0 & \cos(\pi-\theta-\alpha) \end{bmatrix},$$
$$\mathbf{t}(\mathbf{u}) = \begin{bmatrix} \sin(\theta) \\ 0 \\ \cos(\theta) \end{bmatrix}. \tag{1}$$

It should be noted that this model does not handle pure rotation. The pure rotation case being 'simpler', because the epipolar geometry is not existent and therefore a point corresponds to a point and not to a line, we decided to not included in our study. But it could be easily added.

We obtain the essential matrix as $\mathbf{E}(\mathbf{u}) = \mathbf{R}(\mathbf{u})[\mathbf{t}(\mathbf{u})]_\times$ and the fundamental matrix as $\mathbf{F}(\mathbf{u}) = \mathbf{K}_\mathcal{T}^{-\top}\mathbf{E}(\mathbf{u})\mathbf{K}_\mathcal{S}^{-1}$. The essential matrix is parametrized by $\mathbf{u} \in [0, 2\pi]^2$. It is possible to determine if a given $\mathbf{u}$ creates overlapping views *i.e.*, if the cameras share a part of their fields of view. The overlapping property $\psi(\mathbf{u})$ is:

Property 1 *Overlapping Property – See supplementary material for Proof*

$$\psi(\mathbf{u}) = 0 \quad \Leftrightarrow \quad \pi + \gamma_\mathcal{S} + \gamma_\mathcal{T} < \theta + \alpha < 3\pi - \gamma_\mathcal{S} - \gamma_\mathcal{T}$$

with $\gamma_\mathcal{S}$ (resp. $\gamma_\mathcal{T}$) half the field of view of the source camera (resp. target). $\psi(\mathbf{u}) = 0$ means no overlap. $\gamma_\mathcal{S}$ (resp. $\gamma_\mathcal{T}$) is extracted from the camera intrinsics in $\mathbf{K}_\mathcal{S}$ (resp. $\mathbf{K}_\mathcal{T}$).

Discretization. We discretize $\mathbb{E}$ using a quad-tree with a root quad $\square_0 = \square\left(\mathbf{0}, [2\pi, 2\pi]^\top\right)$. The subdivision of a level to a finer resolution is performed by splitting all quads of the current level in four quads of equal size. Using the overlapping property one can decide whether or not a quad has to be subdivided. The overlapping indeed has a transitivity property that guarrantees that we do not miss sought solution:

Property 2 *Transitivity of Overlapping Property – See supplementary material for Proof*

$$\begin{aligned} &\textit{Let } \square(\mathbf{a},\mathbf{b}) \subset \square_0 \quad \left(\forall \mathbf{u} \in \mathbf{c}_{\square(\mathbf{a},\mathbf{b})},\, \psi(\mathbf{u}) = 0\right) \\ &\qquad\qquad \Rightarrow \quad \left(\forall \mathbf{u} \in \square(\mathbf{a},\mathbf{b}),\, \psi(\mathbf{u}) = 0\right) \end{aligned}$$

This property means that if none of the 4 corners of a quad satisfies the overlapping property then all the quads issued from this quad can be discarded. This way we can obtain a fine discretization of $\mathbb{E}$ while avoiding to subdivide where it is not of use. We define the h-th layer of subdivision $\mathcal{E}_h$ as the centers of all the quads of the layer, with $\mathcal{E}_0 = \mathbf{o}_{\square_0}$. A result of the subdivision is shown in figure 1.

2.2. Geometric Only Matching

We introduce a method to obtain putative matches based only on geometric constraints. These matches depend only on $\mathbf{u} \in \mathcal{E} \subset \mathbb{R}^n$ and $\mathbf{E}(\mathbf{u}) \in \mathbb{E}$. First a superset of matches is estimated using a loose epipolar constraint from $\mathbf{u}$. The subset which is the most homogeneous from an epipolar point of view is then extracted. Finally, the unicity constraint is enforced. The final set can then be plugged into a robust cost function.

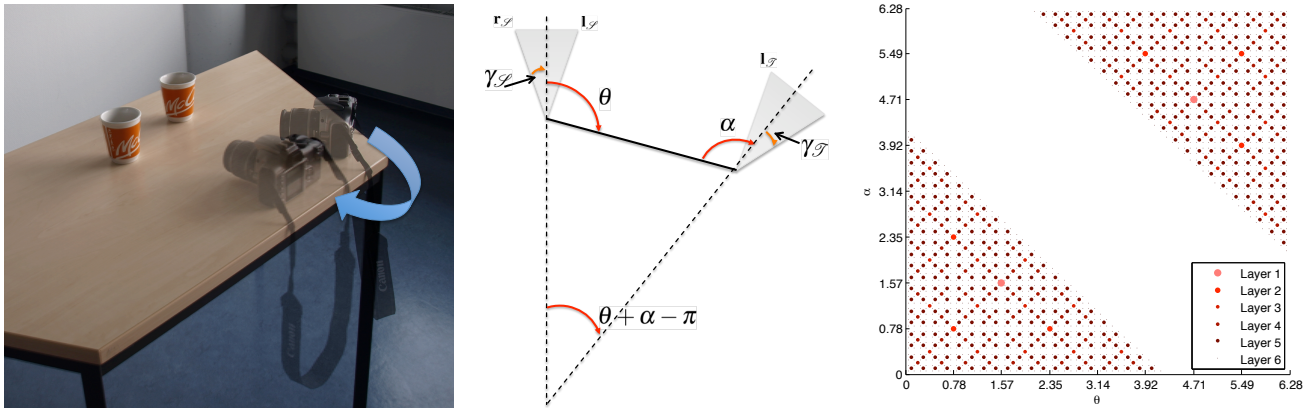

Figure 1. (left) Exemplary in-plane motion (middle) Top view of the 2-parameter camera setup we consider with $\mathbf{u} = [\theta, \alpha]^\top$. The translation only depends on the angle θ while the rotation uses both angles θ and α. (right) A quad-tree subdivision to 6 layers of the essential matrix space $\mathbb{E}$ for the camera setup we consider. The empty diagonal comes from the non-overlapping criterion.

2.2.1 Putative Matches from Geometric Guided Matching

First, we need to define a superset of plausible matches using $\mathbf{E}(\mathbf{u})$. This superset includes the correct matches if $\mathbf{u}$ is the correct solution. One could use a threshold on the distance between a point and an epipolar line but that would give a bias towards the epipole. Therefore we use covariance propagation as suggested in [13] for the propagation of error in the sampled fundamental matrix. Blind PnP [12] uses a similar approach but their trust area is always closed (ellipsoids) making the process less ambiguous. The propagation process defines two hyperbola around the epipolar lines: they define the trust region. The conic equation is as follow:

$$\mathbf{C_q} = \mathbf{l}\mathbf{l}^\top - k^2 \left(\mathbf{J_u} \Sigma_\mathbf{u} \mathbf{J_u}^\top + \mathbf{J_q} \Sigma_\mathbf{q} \mathbf{J_q}^\top \right) \tag{2}$$

with $\mathbf{l} = \mathbf{F}\mathbf{q}$; *i.e.* the epipolar line for point $\mathbf{q}$ and define $\mathbf{J_u}$ (resp. $\mathbf{J_q}$) is the jacobian of $\frac{\mathbf{F}(\mathbf{u})\mathbf{q}}{\|\mathbf{F}(\mathbf{u})\mathbf{q}\|}$ with respect to $\mathbf{u}$ (resp. $\mathbf{q}$). We suppose that the error for the location of point $\mathbf{q}$ and the parameters $\mathbf{u}$ is uniform and therefore $\Sigma_\mathbf{q} = \Sigma_\mathbf{u} = \mathbf{I}$. The only parameter to adjust is k^2 which is in relation with the quality of the results; its choice is discussed later. The criterion of $\langle \mathbf{q}, \mathbf{p} \rangle$ being or not a plausible match is defined as:

$$\begin{aligned} & \sigma_{k^2}(\mathbf{q}, \mathbf{p}, \mathbf{u}) = 1 \\ \Leftrightarrow \quad & \mathbf{p} \in \mathbf{C_q} \\ \Leftrightarrow \quad & \left(\mathbf{p}^\top \mathbf{C_q} \mathbf{p} \right) . \left(\mathbf{p_l}^\top \mathbf{C_q} \mathbf{p_l} \right) > 0 \end{aligned} \tag{3}$$

with $\mathbf{p_l} \in \mathbf{l}$ an arbitrary point on the line $\mathbf{l}$. It is important to note that the test is performed both ways, reverting the role of the source and the target cameras. For simplicity of notation we did not include it. For more information about this guided matching method the reader is referred to [13]. Using test (3) we obtain a rough superset of plausible matches 'around' the epipolar geometry $\mathbf{u}$. It is ambiguous because the epipolar constraint does not predict the point but instead indicates whether a point is compatible with a neighboring epipolar geometry. Therefore we need to cluster this superset, and find an homogeneous, from an epipolar point of view, subset that should mostly contain correct matches.

2.2.2 Extraction of a Coherent Epipolar Geometry by Spectral Clustering

In order to cluster the superset computed in the previous section, the most prominent (*i.e.*, for which the support is the largest) epipolar geometry is selected. We compute a criterion of homogeneity for each pair of matches. This criterion is based on the epipolar geometry $\widetilde{\mathbf{u}}_{kl} \in \mathbb{E}$ that is the closest to $\mathbf{u}$ and verifies the epipolar constraint for $\langle \mathbf{q}_k, \mathbf{p}_l \rangle$. The epipolar geometry $\widetilde{\mathbf{u}}_{kl}$ is defined as:

$$\widetilde{\mathbf{u}}_{kl} = \arg\min_{\widetilde{\mathbf{u}}} d^2_\mathbb{E}(\mathbf{u}, \widetilde{\mathbf{u}}) \quad \text{s.t.} \quad \mathbf{p}_l^\top \mathbf{F}(\widetilde{\mathbf{u}}) \mathbf{q}_k = 0, \tag{4}$$

where $d_\mathbb{E}$ is a pseudo-distance on the variety $\mathbb{E}$ defined in appendix A, different from the euclidean distance. The euclidean distance supposes the variety to be linear in the parameter space. However, $\mathbb{E}$ is nonlinear, as (1) shows. Our distance $d_\mathbb{E}$ represents more accurately the underlying nonlinear variety $\mathbb{E}$. (4) is solved using Lagrange multipliers.It should be noted that the resulting $\widetilde{\mathbf{u}}_{kl}$ might not belong to the discrete set $\mathcal{E}$.

Using the estimated parameters $\widetilde{\mathbf{u}}$ we build a similarity matrix $\mathbf{S}$, that compares two epipolar geometries $\widetilde{\mathbf{u}}_{kl}$ and $\widetilde{\mathbf{u}}_{fg}$, as follows:

$$\begin{cases} \mathbf{S}(i,j) & = \rho_{4\tau} \left(\begin{array}{ll} & d_l(\mathbf{q}_k, \mathbf{F}(\mathbf{u}_{fg})^\top \mathbf{p}_l) \\ + & d_l(\mathbf{q}_f, \mathbf{F}(\mathbf{u}_{kl})^\top \mathbf{p}_g) \\ + & d_l(\mathbf{p}_l, \mathbf{F}(\mathbf{u}_{fg}) \mathbf{q}_k) \\ + & d_l(\mathbf{p}_g, \mathbf{F}(\mathbf{u}_{kl}) \mathbf{q}_f) \end{array} \right) \\ \mathbf{S}(j,i) & = \mathbf{S}(i,j) \\ \mathbf{S}(i,i) & = 0 \end{cases} \tag{5}$$

with ρ_τ a kernel function with a compact support of size τ (see supplementary materials for details) and i (resp. j) corresponds to the matrix entry for the pair $\langle \mathbf{q}_k, \mathbf{p}_l \rangle$ (resp. $\langle \mathbf{q}_f, \mathbf{p}_g \rangle$).

Now we form the normalized Laplacian matrix $\mathbf{L} = \mathbf{D}^{-\frac{1}{2}} (\mathbf{D} - \mathbf{S}) \mathbf{D}^{-\frac{1}{2}}$ with $\mathbf{D}$ the diagonal matrix composed of $\sum_j \mathbf{S}(i,j)$ as explained in [16] and perform an eigenvalue decomposition of $\mathbf{L}$. The smallest (non-zero) eigenvector clusters the space between a set of compatible matches and a set of heterogenous matches. The cut is made for values greater than $\tau_{eig} = \frac{1}{\sqrt{2 n_{Corres}}}$. This threshold τ_{eig} is based on the fact that the eigenvector is a unit vector of dimension n_{Corres} and based on our experiments it performs well for this problem. This gives a subset of homogeneous (from an epipolar point of view) matches but it might include points which are matched multiple times, which is not feasible. Unicity has to be enforced to obtain a usable set of matches for the final robust estimation step.

2.2.3 Enforcing Unicity

Once we have a set of homogeneous matches $\mathcal{M}$ we have to enforce the unicity constraint for the matches. We do not want to have a point in $\mathcal{S}$ (resp. $\mathcal{T}$) to be matched several times in $\mathcal{T}$ (resp. $\mathcal{S}$) because it would bias the robust cost function. We create Ω the set of set of matches such that unicity is satisfied for all sets $\widehat{\mathcal{M}}_j \in \Omega$. These sets verify:

$$\forall \langle \mathbf{q}, \mathbf{p} \rangle \in \widehat{\mathcal{M}}_j \text{ s.t. } \begin{cases} \langle \mathbf{q}, \mathbf{p} \rangle \in \mathcal{M} \\ \nexists \mathbf{q}' \neq \mathbf{q} \text{ s.t. } \langle \mathbf{q}', \mathbf{p} \rangle \in \widehat{\mathcal{M}}_j \\ \nexists \mathbf{p}' \neq \mathbf{p} \text{ s.t. } \langle \mathbf{q}, \mathbf{p}' \rangle \in \widehat{\mathcal{M}}_j \end{cases} \quad (6)$$

We determine the best set in Ω as:

$$\widehat{\mathcal{M}} = \arg \max_{\widehat{\mathcal{M}}^j \in \Omega} \sum_k \mathbf{D}_k, \quad (7)$$

with $\mathbf{D}$ computed as in section 2.2.2. The maximum score represents the largest set which is the most homogeneous / similar (remember that $\mathbf{S}$ is a similarity matrix) in comparison to the others.

We finally obtain a coherent and unicity-satisfaying set of $\widehat{\mathcal{M}}$ matches 'around' $\mathbf{u}$. We determine the parameter $\widehat{\mathbf{u}}$ that best represent $\widehat{\mathcal{M}}$ using the similarity matrix $\mathbf{S}$ as:

$$\widehat{\mathbf{u}} = \mathbf{u}_i \quad \text{s.t.} \quad i = \arg \max_j \mathbf{D}_{j,j} \quad (8)$$

2.3. Nonlinear Motion Refinement

Using a set of putative matches we still have to estimate an essential matrix accurately. We cannot guarantee that the putative matches only include inliers. Therefore we have to use robust nonlinear estimation based on the point to epipolar line cost function. We propose to minimize the following cost:

$$q(\mathbf{u}) = \sum_{\langle \mathbf{q}, \mathbf{p} \rangle \in \widehat{\mathcal{M}}} \left(\begin{array}{cc} & \left(1 - \rho_\tau \left(d_l \left(\mathbf{p}, \mathbf{F}(\mathbf{u}) \mathbf{q}\right)\right)\right)^2 \\ + & \left(1 - \rho_\tau \left(d_l \left(\mathbf{q}, \mathbf{F}^\top(\mathbf{u}) \mathbf{p}\right)\right)\right)^2 \end{array} \right) \quad (9)$$

with ρ_τ a compact kernel (see supplementary materials). We use $\widehat{\mathbf{u}}$ for the initialization of the minimization. Our cost function q disregards those points which are far off the epipolar lines (by more than τ pixels).

The different steps of the algorithm are summarized in 1.

Input: Subdivisions $\{\mathcal{E}_h\}$, keypoints $\{\mathbf{q}_k\}$ in the source and $\{\mathbf{p}_l\}$ in the target images
Output: Solutions `sol` composed of an epipolar geometry $\widetilde{\mathbf{u}}$ and pairs of matches $\widetilde{\mathcal{M}}$
foreach $\mathcal{E} \in \{\mathcal{E}_h\}$ **do**
  foreach $\mathbf{u} \in \mathcal{E}$ **do**
    foreach $\langle \mathbf{q}_k, \mathbf{p}_l \rangle$ **such that** $\sigma_{k^2}(\mathbf{q}_k, \mathbf{p}_l, \mathbf{u}) = 1$ **do**
      $\mathcal{M}$.push-back($\langle \mathbf{q}_k, \mathbf{p}_l \rangle$);
      compute $\mathbf{u}_{kl}$ using (4) with $\langle \mathbf{q}_k, \mathbf{p}_l \rangle$;
    end
    if $\mathcal{M}$.size() $\geqslant n_{min}$ **then**
      $\widetilde{\mathcal{M}} \leftarrow$ cluster $\mathcal{M}$ % Section 2.2.2;
      $\widehat{\mathcal{M}} \leftarrow$ enforce unicity on $\widetilde{\mathcal{M}}$ % Section 2.2.3;
      if $\widehat{\mathcal{M}}$.size() $\geqslant n_{min}$ **then**
        compute $\widetilde{\mathbf{u}}$ using $\left\langle \widehat{\mathbf{u}}, \widehat{\mathcal{M}} \right\rangle$ % Section 2.3;
        if $RE(\widetilde{\mathbf{u}}, \widehat{\mathcal{M}}) < \tau_{noise}$ **then**
          sol.push-back($\left\langle \widetilde{\mathbf{u}}, \widehat{\mathcal{M}} \right\rangle$);
        end
      end
    end
    if sol.size() > 0 **then**
      return;
    end
  end
end

Algorithm 1: Proposed algorithm for simultaneous camera motion estimation and point matching.

3. Experiments and Results

In this section we describe the experiment setup, implementation details and present results on synthetic and real data.

3.1. Implementation Details

Algorithm 1 was implemented in Matlab. The constrained estimation (4) was solved using `fmincon` and the robust cost (9) was minimized using `lsqnonlin`. If the algorithm finds several minima, we select the solution with maximum number of matches. All the experiments were done on different quad-core machines with the use of `parfor` (parallel for loop) to easily spread the calculation between cores. The synthetic data were created using a camera with a field of view of 30.96°. The 3D points were generated randomly in front of the cameras and then projected to the views. The minimum number of necessary matched points n_{min} was 8.

All the estimated parameter $\widetilde{\mathbf{u}}$ were evaluated against the estimated pair of matches $\widetilde{\mathcal{M}}$ and the true pair of matches

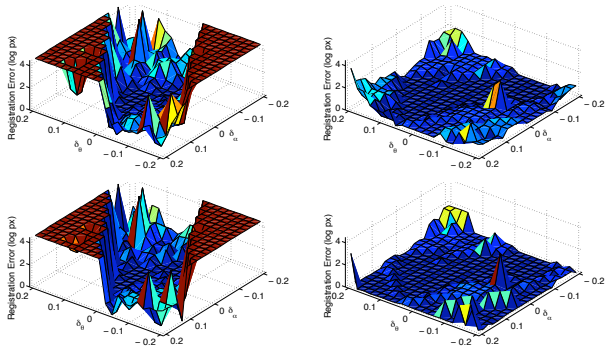

Figure 2. [up] Log Registration Error (RE) results obtained with estimated matches with varying starting point RE($\widetilde{\mathbf{u}}, \widetilde{\mathcal{M}}$) [down] Log RE results with true matches RE$\widetilde{\mathbf{u}}, \overline{\mathcal{M}}$) (left) $k^2 = 0.01$, $\tau = 10$ (right) $k^2 = 0.102$, $\tau = 15$.

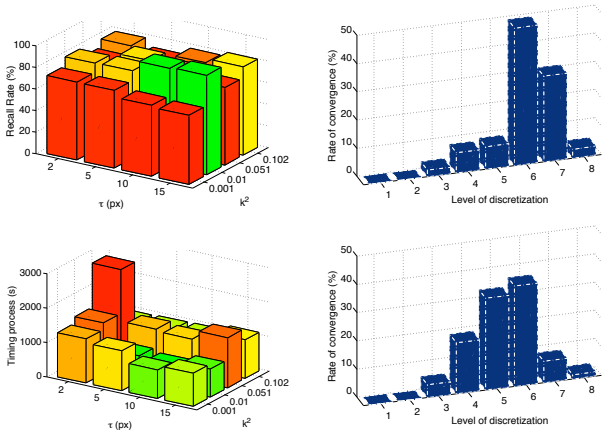

Figure 3. [left] Impact of the variation of k^2 and τ (top) Recall rate (bottom) Mean processing time; [right] Histogram of the level of discretization on which the algorithm converged (top) $k^2 = 0.01$, $\tau = 10$ (bottom) $k^2 = 0.102$, $\tau = 15$.

$\overline{\mathcal{M}}$ with a point to line Registration Error model :

$$\mathrm{RE}\left(\mathbf{u}, \mathcal{M}\right) = \frac{1}{2N} \sum_{i=0}^{N} \left(\begin{array}{l} d_l\left(\mathbf{F}\left(\mathbf{u}\right)\mathbf{q}_i, \mathbf{p}_i\right) \\ + \quad d_l\left(\mathbf{F}^{\top}\left(\mathbf{u}\right)\mathbf{p}_i, \mathbf{q}_i\right) \end{array} \right). \tag{10}$$

3.2. Experiments

First, we want to determine a correct value for the inverse of the chi-square cumulative distribution (k^2) and a threshold τ used for the similarity and for the robust cost function (9). The former impacts the size of the putative matches sets and therefore the computational time. The latter modifies the convergence and stability against noise. They both change the basin of convergence. As shown in figure 2 where we applied the matching algorithm using as input parameter $\mathbf{u}_0$ around $\overline{\mathbf{u}}$ (with 25 3D points). We varied the parameters in $[-0.2;\, 0.2]^2$. In figure 2, the reader can get a feeling of the convergence rate evolution depending on k^2 and τ . Furthermore, it should be noted that the final registration error applied to the estimated matches $\widetilde{\mathcal{M}}$ and the known matches $\overline{\mathcal{M}}$ give similar numerical result. This means that when the algorithm converged it is toward the true solution.

Then we drew random parameters $\overline{\mathbf{u}}$ and a structure of 25 3D points to perform the estimation (without noise and outlier); this means that we have 25 source points that have a match in the target image. We used the complete algorithm over eight discretization level. All process converged to a pose which registered the true match ($RE\left(\widetilde{\mathbf{u}}, \overline{\mathcal{M}}\right)$) under an error of 10^{-10}. We study the variation of the processing time and recall rate (percentage of matches found) and the mean processing time. Some combination gave interesting results with good recall rates (more $> 80\%$) and interesting speed: $\left\{k^2, \tau\right\} \in \left\{\{0.01, 10\}, \{0.01, 15\}, \{0.102, 15\}\right\}$. It is interesting to see that the smaller the superset of matches is (when k_2 is small), the more the convergence happens on at precise (*e.g.* higher) discretization, see figure 3. We decide to use $\left\{k^2, \tau\right\} = \{0.01, 15\}$ because it gave a good recall rate and was offering the best possible speed. For the rest of the experiments we used only 7 subdivisions (2502 essential matrices) because its offers a good convergences ($> 95\%$) and speeds up the process compare to an 8^{-th} subdivision which adds 7482 new configurations to test.

Then, we studied the consequence of clutter for the performance of the algorithm. We used again 25 3D points. Depending on the required number of outliers we removed true matches and replaced them by random points. We used $\sigma_{noise} = 10^{-10}$ pixels. For each percentage of outliers we used 50 randomly picked poses. We studied the true convergence rate, the false convergence rate (percentage of result where we thought we converged but it was wrong), the computation time, the recall rate (how many percents of the true matches we found) and finally the percentage of false/erroneous matches (estimated matches that were not present in the true matches set). We always computed a average over all runs. The results are visible in figure 4(a). The first lesson about this experiment is that we never converge to a false minimum: we either converge to the right solution or we diagnose we did not. Second the processing time can quadruple with increasing clutter. The main reason for this is that we have to process through $\mathcal{E}$ before declaring that we did not find any solution. Third, the matches are well extracted from the data with a mean recall rate over the different levels of clutter of 75 %. Finally, there are few to no false matches. Overall we see that the method performs well in the presence of clutter.

Thereafter, we performed experiments against noise in the detected points. We applied the same protocol as for the previous experiments and corrupt the image points with a Gaussian noise of standard deviation σ. We decide upon

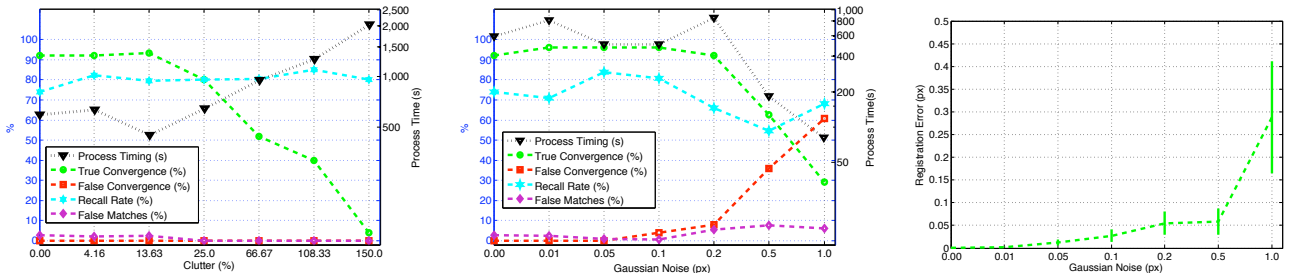

Figure 4. (left) Experiments against clutter: the proposed method has a good convergence and recall rate;(center) Experiments against feature detection noise: again, the proposed method performs well; (right) Registration error after convergence. (Note: time in log scales)

convergence based on the noise level $\sigma_{noise} = \sqrt{2}\sigma$. The results are displayed in figure 4(b). This demonstrates that our algorithm handles noise even though we only consider 25 3D points. The algorithm performs faster with increasing noise because the increasing convergence threshold and therefore the convergence condition becomes looser. That is why the rate of wrong convergence increases with the noise because the criteria is too loose. The performance of the algorithm degrades with the noise level but offers good registration error as shown in figure 4(c). We speculate that using a re-projection error cost would improve this result since it is a ML estimator.

Finally, using a calibrated camera on a table we captured a scene from two different views and selected harris corners. The scene is composed of two identical objects that would confuse any photometric based approach. See in figure 5 how our algorithm copes even in the presence of geometric ambiguities (geometrically aligned structured). It manage to match only one of the cup because of the unprecision of the keypoints and calibration. The resulting pose is $\widetilde{\mathbf{u}} = [114.3280, 93.9593]^\top$ (angles are expressed in degrees) which is close to the manually estimated pose $\overline{\mathbf{u}} = [113.2165, 94.9960]^\top$. This demonstrates the applicability of the method with real data.

4. Conclusion

We presented a framework to estimate the epipolar geometry from two unmatched sets of keypoints without using photometric information. When RANSAC-like approaches use matches to simultaneously estimate the motion, we estimate one motion and the matches. In order to achieve this difficult task, we use a discrete set of essential matrices and an efficient and powerful geometric only matching procedure. The variety of essential matrices is explored using a proposed pseudo-distance between essential matrices. The matching method makes use of error propagation for guided matching, spectral clustering with a new similarity measure to extract homogeneous matches and ensures matches to be unique by post processing. Finally, the obtained geometric putative matches are evaluated through robust nonlinear estimation. We demonstrated the use of the algorithm on a moving camera mounted on a tripod that bares all the ambiguities of epipolar geometry but not its dimensionally burden.

Future work should include a general overlapping criterion and transitivity property for the full 5 degrees of freedoms case. It should also include the use of quads to evaluate the covariance of the final estimate. The implementation should make more use of the parallel aspect of the approach.

Acknowledgement: This work was made possible by DAAD and Egide which funded the travel grant Surf-3D.

A. Fundamental and Essential Distances

In this appendix, we introduce a pseudo-distance between epipolar geometries. The proposed measure is based on the distance introduced by Zhang [18], though our distance is more systematic since it does not include a random process.

In order to introduce our distance we define a set of tools. Let $\mathbf{F}_1$ and $\mathbf{F}_2$ be the two fundamental matrices wg=hich define the two epipolar geometries to compare. We define as $\mathcal{C}_\mathcal{T}$ the conic that passes through the four corners of the image $\mathcal{T}$. We will not be computing explicitly these conics. Instead we define the transformation $\mathbf{T}_\mathcal{T}$ between the unit circle $\mathcal{C}^\mathcal{O}$ and $\mathcal{C}^\mathcal{T}$; this can be trivialy estimated. Let $\mathbf{q}$ be an image point in $\mathcal{S}$ and $\mathbf{l}_i = \mathbf{F}_i\mathbf{q}$ (with $i \in \{1, 2\}$) its corresponding epipolar line in $\mathcal{T}$. To improve the readability we define $\mathbf{l}^\mathcal{T} := \mathbf{T}_\mathcal{T}^{-\top}\mathbf{l}$ as an epipolar line in the coordinate frame of $\mathcal{C}^\mathcal{O}$. We define $\cap^\mathcal{T}(\mathbf{l}) = \mathbf{l} \cap \mathcal{C}^\mathcal{T}$ to be the intersection between the epipolar line and the conic of $\mathcal{T}$. In order to simplify the calculation we compute the intersection within the coordinate system of the unit circle $\mathcal{C}^\mathcal{O}$, therefore we rewrite the intersection problem as $\cap^\mathcal{T}(\mathbf{l}) = \mathbf{T}_\mathcal{T}^{-1}\left(\mathbf{l}^\mathcal{T} \cap \mathcal{O}\right)$.

Three results are possible: no intersection, one intersection point, two intersection points. These different cases will be sorted using the distance $d_\mathcal{O}$ from a line to the origin $\mathbf{l}$. The 'no intersection' case is determined when $d_\mathcal{O}\left(\mathbf{l}^\mathcal{T}\right) > 1$. The two other cases happen for $d_\mathcal{O}\left(\mathbf{l}^\mathcal{T}\right) \leq 1$.

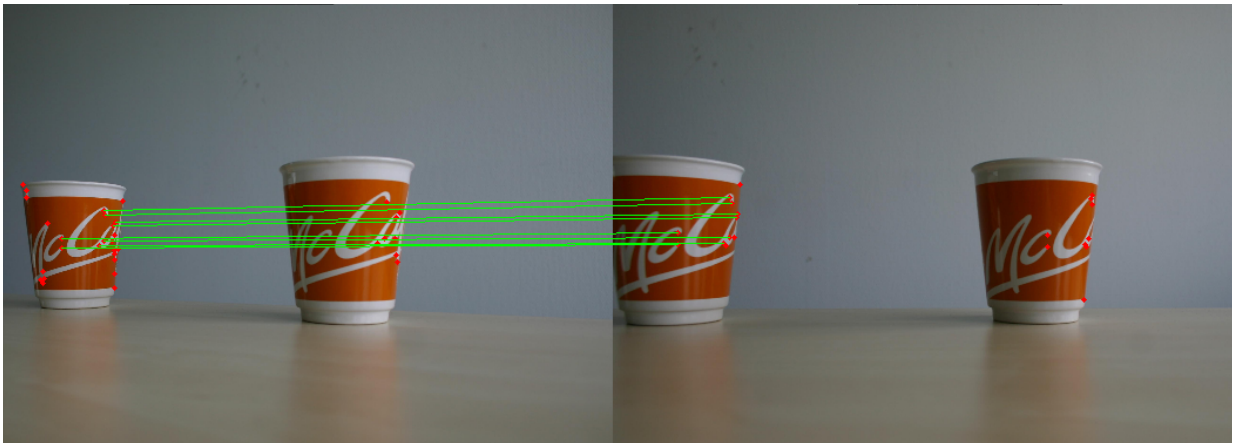

Figure 5. Wide Baseline Experiment; Matching result with multiple similar object; most of the features are repeated between the two cups; in green correct matches and in red the input keypoints.

Since they can be summarized in the same equation, we write $\cap^{\mathcal{T}} := \cap^{\mathcal{T}}_{\cdot}(\mathbf{l})$, with $\cdot \in \{\pm, \mp\}$. Expressed in the coordinate system of $\mathcal{O}$, the solutions of $\mathbf{l}^{\mathcal{T}} \cap \mathcal{O}$ are:

$$d_{\mathcal{O}}\left(\mathbf{l}^{\mathcal{T}}\right) \leq 1 \Rightarrow \mathbf{l}^{\mathcal{T}} \cap \mathcal{O} = w\left(\begin{bmatrix} -l_1 l_3 \pm l_2 r\left(\mathbf{l}^{\mathcal{T}}\right) \\ -l_2 l_3 \mp l_1 r\left(\mathbf{l}^{\mathcal{T}}\right) \\ l_1^2 + l_2^2 \end{bmatrix}\right), \tag{11}$$

with w the projection function $([x/z,\, y/z,\, 1]^{\top})$ and $r\left(\mathbf{l}^{\mathcal{T}}\right) = \sqrt{l_1^2 + l_2^2 - l_3^2}$.

We are now ready to define our distance function. We discretize the image $\mathcal{S}$ into a set of 2D points $\mathbf{q} \in \mathcal{S}$ and define:

$$d_{\mathbb{F}}\left(\mathbf{F}_1, \mathbf{F}_2\right) = \sum_{\mathbf{q} \in \mathcal{S}} \sigma\left(\mathbf{l}_1\right) \sigma\left(\mathbf{l}_2\right) \sum_{\cdot}^{\{\pm,\mp\}} \left\| \overset{\mathcal{T}}{\underset{\cdot}{\cap}}\left(\mathbf{l}_1\right) - \overset{\mathcal{T}}{\underset{\cdot}{\cap}}\left(\mathbf{l}_2\right) \right\| \tag{12}$$

with $\sigma\left(\mathbf{l}_i\right) = s\left(d_{\mathcal{O}}\left(\mathbf{l}_i^{\mathcal{T}}\right)\right)$ and s representing a steep sigmoid that is truncated to zero for all points that do not cross the conic (*i.e.* > 1); this sigmoid is used to have a continuous behavior.

In order to obtain a symmetric function we redefine (12) by: $d_{\mathbb{F}}\left(\mathbf{F}_1, \mathbf{F}_2\right) \leftarrow d_{\mathbb{F}}\left(\mathbf{F}_1, \mathbf{F}_2\right) + d_{\mathbb{F}}\left(\mathbf{F}_1^{\top}, \mathbf{F}_2^{\top}\right)$.

We can finally define the essential distance $d_{\mathbb{E}}$ and for simplicity of notation we write $d_{\mathbb{E}}\left(\mathbf{u}_1, \mathbf{u}_2\right) = d_{\mathbb{F}}\left(\mathbf{K}_{\mathcal{T}}^{-\top} \mathbf{E}\left(\mathbf{u}_1\right) \mathbf{K}_{\mathcal{S}}^{-1}, \mathbf{K}_{\mathcal{T}}^{-\top} \mathbf{E}\left(\mathbf{u}_2\right) \mathbf{K}_{\mathcal{S}}^{-1}\right)$.

References

[1] H. Bay, A. Ess, T. Tuytelaars, and L. van Gool. Speeded-up robust features (SURF). *CVIU*, 2008.

[2] T. M. Breuel. A Practical, Globally Optimal Algorithm for Geometric Matching under Uncertainty. *Electronic Notes in Theoretical Computer Science*, July 2001.

[3] O. Chum, T. Werner, and J. Matas. Epipolar geometry estimation via ransac benefits from the oriented epipolar constraint. *ICPR*, 1, 2004.

[4] F. Dellaert, S. M. Seitz, C. E. Thorpe, and S. Thrun. Structure from motion without correspondence. In *CVPR*, 2000.

[5] M. Fischler and R. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Com. of ACM*, 1981.

[6] R. Hartley. In Defence of the 8-Point Algorithm. *ICCV*, 1995.

[7] R. Hartley and F. Kahl. Global Optimization through Searching Rotation Space and Optimal Estimation of the Essential Matrix. In *ICCV*, October 2007.

[8] T. Huang and O. Faugeras. Some properties of the E matrix in Two-View Motion Estimation. *IEEE PAMI*, 1989.

[9] D. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *IJCV*, 2004.

[10] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Gool. A Comparison of Affine Region Detectors. *IJCV*, 2005.

[11] L. Moisan and B. Stival. A probabilistic criterion to detect rigid point matches between two images and estimate the fundamental matrix. *IJCV*, 2004.

[12] F. Moreno-Noguer, V. Lepetit, and P. Fua. Pose priors for simultaneously solving alignment and correspondence. *ECCV*, January 2008.

[13] B. Ochoa and S. Belongie. Covariance Propagation for Guided Matching. *Workshop on Statistical Methods in Multi-Image and Video Processing*, 2006.

[14] A. Rangarajan, H. Chui, and F. L. Bookstein. The Softassign Procrustes Matching Algorithm. *Information Processing in Medical Imaging*, April 1999.

[15] H. Stewenius, C. Engels, and D. Nister. Recent developments on direct relative orientation. *Journal of Photogrammetry and Remote Sensing*, 2006.

[16] U. von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 2007.

[17] H. Wolfson and I. Rigoutsos. Geometric hashing: An overview. *IEEE Computational Science & Engineering*, 1997.

[18] Z. Zhang. Determining the Epipolar Geometry and its Uncertainty: A Review. *IJCV*, 1998.