1-Point RANSAC for EKF-Based Structure from Motion

Javier Civera, Oscar G. Grasa, Andrew J. Davison and J. M. M. Montiel

Abstract— **Recently, classical pairwise Structure From Motion (SfM) techniques have been combined with non-linear global optimization (Bundle Adjustment, BA) over a sliding window to recursively provide camera pose and feature location estimation from long image sequences. Normally called Visual Odometry, these algorithms are nowadays able to estimate with impressive accuracy trajectories of hundreds of meters; either from an image sequence (usually stereo) as the only input, or combining visual and propioceptive information from inertial sensors or wheel odometry.**

This paper has a double objective. First, we aim to illustrate for the first time how similar accuracy and trajectory length can be achieved by filtering-based visual SLAM methods. Specifically, a camera-centered Extended Kalman Filter is used here to process a monocular sequence as the only input, with 6DOF motion estimated. Features are kept live in the filter while visible as the camera explores forward, and are deleted from the state once they go out of view.

This permits an increase in the number of tracked features per frame from tens to around a hundred. While improving the accuracy of the estimation, it makes computationally infeasible the exhaustive Branch and Bound search performed by standard JCBB for match outlier rejection. As a second contribution that overcomes this problem, we present here a RANSAC-like algorithm that exploits the probabilistic prediction of the filter. This use of prior information makes it possible to reduce the size of the minimal data subset to instantiate a hypothesis to the minimum possible of 1 point, greatly increasing the efficiency of the outlier rejection stage.

Experimental results from real image sequences covering trajectories of hundreds of meters are presented and compared against RTK GPS ground truth. Estimation errors are about 1% of the trajectory for trajectories up to 650 metres.

I. Introduction

Classical Structure from Motion (SfM) [12] and Bundle Adjustment (BA) [27] techniques have recently been adapted to sequential and real-time processing of long image sequences. Real-time performance has been achieved by performing live optimization over only a limited number frames of the sequence. If these frames are chosen to be 'keyframes' sparsely distributed around a working volume, this permits accurate and drift-free room-sized mapping as in [13]. Or in the case we are considering in this paper, choosing a sliding window of the most recently acquired frames permits accurate camera motion estimation for long trajectories (e.g. [16] for a monocular camera). These latter approaches are generically known as Visual Odometry, and it has also been demonstrated that the trajectory estimates they produce can be combined with loop closure techniques to construct large but consistent maps (e.g. [26] or [14] with stereo vision).

The front-end image processing algorithms in all of the previously mentioned approaches are similar. In a few words, salient features are extracted and correspondences are searched for between a window of currently live frames. Scene structure and camera poses are estimated for the selected live frames via standard SfM methods. Finally, the solution is refined in a Bundle Adjustment optimization step.

Sequential 3D motion and structure estimation from a monocular camera has also been tackled using filtering schemes [1], [4], [9] which propagate a probabilistic state estimate. Accurate estimation for large trajectories have only been achieved after the addition of other estimation techniques, like local mapping and loop closing in [8], [10], [20], [22].

The first aim of this paper is to show how filtering algorithms can reach similar accuracy to current Visual Odometry methods. To achieve this, we used the sensor-centered Extended Kalman Filter, introduced first in the context of laser-based SLAM [3]. Contrary to the standard EKF, where estimates are always referred to a world reference frame, the sensor-centered approach represents all feature locations and the camera motion in a reference frame local to the current camera. The typical correlated feature-sensor uncertainty arising from the uncertain camera location is transferred into uncertainty in the world reference frame, resulting in lower variances for the camera and map features, and thus in smaller linearization errors.

Another key difference of our approach when compared against usual EKF visual SLAM is the number of measured features, which we increase from tens to more than a hundred (see figure 1). It has been observed experimentally that this increase highly improves the accuracy of the estimation and also makes scale drift, previously reported in [4], [8], [20], almost vanish for the trajectory lengths considered.

The second contribution of this paper is the proposal of a new RANSAC algorithm that exploits the probabilistic prediction obtained from the EKF in order to increase the efficiency of the spurious match rejection step. This need for an increase in the efficiency is motivated by the high computational cost of the Joint Compatibility Branch and Bound algorithm (JCBB) [17], which specifically is exponential in the number of measurements. Although its use has been proven feasible and advisable in [8] for one or two tens of matches, the algorithm quickly becomes computationally intractable when the number of matches grows near a hundred.

In recent years, an important stream of research has

Javier Civera, Oscar G. Grasa and J. M. M. Montiel are with Instituto de Investigación e Ingeniería de Aragón (I3A), Universidad de Zaragoza, Spain. {jcivera, oscgg, josemari}@unizar.es

Andrew J. Davison is with the Department of Computing, Imperial College, London, UK. ajd@doc.ic.ac.uk

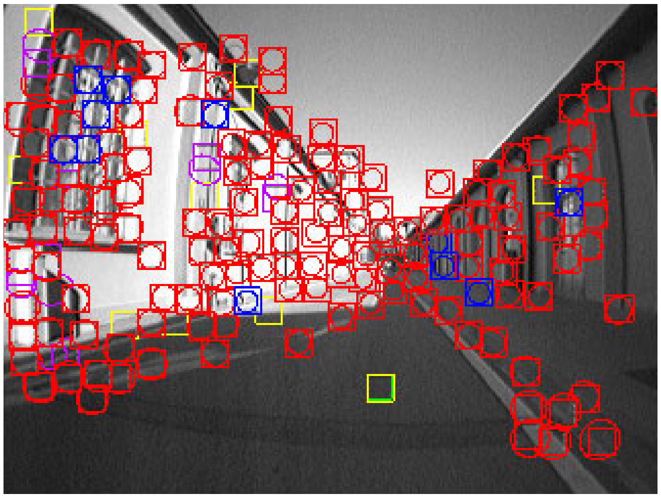

Fig. 1. Example of image from the monocular sequence. Square represents image patches from tracked features; and ellipses show the individual compatibility regions.

focused on reducing the model verification cost in standard RANSAC [11] (e.g. [23], [2], [6]) by the early detection and termination of bad hypotheses. The RANSAC algorithm proposed here is able to reduce the model verification cost by greatly reducing the size of the set of matches needed to instantiate the model to the minimal size of 1. As a consequence of this, the number of random hypothesis necessary to obtain a mismatch-free subset is reduced by several orders of magnitude.

Incorporating probabilistic predictions into RANSAC has been previously investigated in [15] for the case of weak priors and also in an EKF context [28]. Nevertheless, the reduction of the subset size and hence full exploitation of the strong a priori information available in the EKF is not explored in these works. The Active Matching method of [5], which is an information-driven one-by-one matching approach, robustifies and improves the efficiency of correspondence search given a probabilistic prediction, and is the clearest inspiration for our method: it shows how integrating the first match highly constrains the possible image locations of other features. This fact is used in this paper to propose the 1-point hypothesis; which in the case of the EKF with tightly correlated priors is enough to discard spurious matches.

RANSAC using 1-point hypotheses has been also very recently proposed in [25], reporting a similar reduction in the number of random hypotheses that are needed as in our approach. In that paper, however, this reduction comes as a result of incorporating into the motion model some restrictions on the allowed motion; specifically planar motion and a large radius of curvature typical of car motion is assumed. In our case, the extra information available from a motion model to aid matching is dealt with in a much more general manner, and we are able to cope with smooth camera motion with the full six degrees of freedom.

The rest of the paper is organised as follows. We begin by introducing the camera-centered Extended Kalman Filter in section II. In section III the 1-point RANSAC algorithm proposed is fully described. Section IV is devoted to experimental validation of the approach using three sequences extracted from the publicly available dataset [24]. Finally, we conclude in section V and discuss lines of future work in section VI.

II. Camera-centered EKF-based estimation

It is a well-known fact that one of the major problems of the Extended Kalman Filter estimator is the early appearance of inconsistency due to linearization errors. This problem is particularly worrying in our visual odometry case: as every point of the scene is removed from the estimation as soon as it is not seen in the image and revisited places are not detected, the uncertainty in the camera location *always* grows with respect to a world reference frame as the camera moves away from the origin. As the uncertainty grows, the linearizations performed by the EKF are less valid.

To overcome this problem, camera-centered EKF-based estimation taken from [3] is used in this paper. The linearization error reduction that this technique offers is based on the fact that EKF computations are going to be made, in the general case, with elements closer to the camera frame than any other reference. Locking the frame of reference to the camera will lead to lower uncertainties, and linearizations will be more valid.

In our camera-centered representation, the estimation at every step k is parameterized as a multidimensional Gaussian $\mathbf{x}_k \sim \mathcal{N}\left(\hat{\mathbf{x}}_k, \mathbf{P}_k\right)$ that includes the location of the world reference frame $\mathbf{x}_W^C$ as a non-observable feature and the map $\mathbf{y}^C$, both in the current camera reference frame. As a difference from [3], the use of a constant velocity model for the camera motion forces us to also keep in the state vector velocity estimates in the camera frame $\mathbf{x}_v^C$.

$$\hat{\mathbf{x}}_k^{C_k} = \begin{pmatrix} \hat{\mathbf{x}}_W^{C_k} \\ \hat{\mathbf{x}}_v^{C_k} \\ \hat{\mathbf{y}}^{C_k} \end{pmatrix} ; \quad \mathbf{P}_k^{C_k} = \begin{pmatrix} \mathbf{P}_W^{C_k} & \mathbf{P}_{Wv}^{C_k} & \mathbf{P}_{Wy}^{C_k} \\ \mathbf{P}_{vW}^{C_k} & \mathbf{P}_v^{C_k} & \mathbf{P}_{vy}^{C_k} \\ \mathbf{P}_{yW}^{C_k} & \mathbf{P}_{yv}^{C_k} & \mathbf{P}_y^{C_k} \end{pmatrix} . \tag{1}$$

The map $\mathbf{y}^{C_k}$ is composed of n point features $\mathbf{y}_i^{C_k}$ which are parametrized using inverse depth coordinates as detailed in [7]:

$$\hat{\mathbf{y}}^{C_k} = \begin{pmatrix} \hat{\mathbf{y}}_1^{C_k} \\ \vdots \\ \hat{\mathbf{y}}_n^{C_k} \end{pmatrix} ; \quad \mathbf{P}_y^{C_k} = \begin{pmatrix} \mathbf{P}_{y_1}^{C_k} & \cdots & \mathbf{P}_{y_1 y_n}^{C_k} \\ \vdots & \ddots & \vdots \\ \mathbf{P}_{y_n y_1}^{C_k} & \cdots & \mathbf{P}_{y_n}^{C_k} \end{pmatrix} . \tag{2}$$

The velocity state vector $\mathbf{x}_v^{C_k}$ stores linear and angular velocities; and the world reference frame coordinates are represented with a position vector and a quaternion:

$$\hat{\mathbf{x}}_v^{C_k} = \begin{pmatrix} \hat{\mathbf{v}}^{C_k} \\ \hat{\omega}^{C_k} \end{pmatrix} ; \quad \hat{\mathbf{x}}_W^{C_k} = \begin{pmatrix} \hat{\mathbf{r}}_W^{C_k} \\ \hat{\mathbf{q}}_W^{C_k} \end{pmatrix} . \tag{3}$$

At every frame, the estimation evolves in three steps: the usual EKF prediction and update steps, and a final composition step which moves the reference frame from the camera at step $k-1$ to the camera at step k.

A. Prediction Step

For the prediction step at time k, the world reference frame and feature map are kept in the reference frame at time $k-1$ and a new feature that represents the motion of the sensor between $k-1$ and k is added:

$$\hat{\mathbf{x}}^{C_{k-1}}_{k|k-1} = \begin{pmatrix} \hat{\mathbf{x}}^{C_{k-1}}_{W} \\ \hat{\mathbf{x}}^{C_{k-1}}_{v} \\ \hat{\mathbf{y}}^{C_{k-1}} \\ \hat{\mathbf{x}}^{C_{k-1}}_{C_k} \end{pmatrix} \tag{4}$$

$$\mathbf{P}^{C_{k-1}}_{k|k-1} = \begin{pmatrix} \mathbf{P}^{C_{k-1}}_{W} & \mathbf{P}^{C_{k-1}}_{Wv} & \mathbf{P}^{C_{k-1}}_{Wy} & \mathbf{0} \\ \mathbf{P}^{C_{k-1}}_{vW} & \mathbf{P}^{C_{k-1}}_{v} & \mathbf{P}^{C_{k-1}}_{vy} & \mathbf{P}^{C_{k-1}}_{vC_k} \\ \mathbf{P}^{C_{k-1}}_{yW} & \mathbf{P}^{C_{k-1}}_{yv} & \mathbf{P}^{C_{k-1}}_{y} & \mathbf{0} \\ \mathbf{0} & \mathbf{P}^{C_{k-1}}_{C_k v} & \mathbf{0} & \mathbf{Q}^{C_{k-1}} \end{pmatrix} , \tag{5}$$

where $\mathbf{Q}^{C_{k-1}}$ is the covariance of the zero-mean Gaussian acceleration noise.

Camera motion between $k-1$ and k is computed by applying a constant velocity model [9]. $\hat{\mathbf{x}}^{C_{k-1}}_{C_k}$ represents camera location via a position vector and quaternion:

$$\hat{\mathbf{x}}^{C_{k-1}}_{C_k} = \begin{pmatrix} \hat{\mathbf{r}}^{C_{k-1}}_{C_k} \\ \hat{\mathbf{q}}^{C_{k-1}}_{C_k} \end{pmatrix} . \tag{6}$$

Contrary to what might be expected, motion in our sensor-centered estimation scheme is not applied then over the features; it is applied to the camera first and then the reference frame is moved after the update. This postponement of the composition step aims at a further reduction of the linearization error: as the updated covariance is smaller than the predicted one, the linearization of the composition will have less error if performed after the update [3].

B. Update Step

The update is performed using the standard Extended Kalman Filter equations:

$$\begin{aligned} \hat{\mathbf{x}}^{C_{k-1}}_{k} &= \hat{\mathbf{x}}^{C_{k-1}}_{k|k-1} + \mathbf{K}_k \left(\mathbf{z}_k - \mathbf{h}_k \left(\hat{\mathbf{x}}^{C_{k-1}}_{k|k-1} \right) \right) & (7) \\ \mathbf{P}^{C_{k-1}}_{k} &= \left(\mathbf{I} - \mathbf{K}_k \mathbf{H}_k \right) \mathbf{P}^{C_{k-1}}_{k|k-1} & (8) \\ \mathbf{K}_k &= \mathbf{P}^{C_{k-1}}_{k|k-1} \mathbf{H}^{\top}_k \mathbf{S}^{-1}_k , & (9) \end{aligned}$$

where $\mathbf{h}_k$ is the measurement equation, composed of a pinhole camera model plus a radial distortion lens model. $\mathbf{H}_k$ is the Jacobian of this model evaluated at the prediction $\left(\mathbf{H}_k = \frac{\partial \mathbf{h}}{\partial \mathbf{x}^{C_{k-1}}} \Big|_{\mathbf{x}^{C_{k-1}}_{k|k-1}} \right)$ and $\mathbf{S}_k$ is the image covariance, computed as the sum of the propagated state covariance plus the zero-mean image noise covariance $\mathbf{R}_k$ $\left(\mathbf{S}_k = \mathbf{H}_k \mathbf{P}^{C_{k-1}}_{k|k-1} \mathbf{H}^{\top}_k + \mathbf{R}_k \right)$.

C. Composition Step

Finally, in the composition step, a rigid transformation is applied to the world reference and the estimated features that moves them from the previous camera reference frame to the current one. The rigid transformation between the previous frame of reference and the current one is removed from the estimation. The resulting state vector is:

$$\hat{\mathbf{x}}^{C_k}_{k} = \begin{pmatrix} \hat{\mathbf{x}}^{C_k}_{W} \\ \hat{\mathbf{x}}^{C_k}_{v} \\ \hat{\mathbf{y}}^{C_k} \end{pmatrix} , \tag{10}$$

where $\hat{\mathbf{x}}^{C_k}_{W}$, $\hat{\mathbf{x}}^{C_k}_{v}$ and $\hat{\mathbf{y}}^{C_k}$ have been computed by composition with the motion between frames $\hat{\mathbf{x}}^{C_{k-1}}_{C_k}$:

$$\begin{aligned} \hat{\mathbf{x}}^{C_k}_{W} &= \ominus \hat{\mathbf{x}}^{C_{k-1}}_{C_k} \oplus \hat{\mathbf{x}}^{C_{k-1}}_{W} & (11) \\ \hat{\mathbf{x}}^{C_k}_{v} &= \ominus \hat{\mathbf{x}}^{C_{k-1}}_{C_k} \oplus \hat{\mathbf{x}}^{C_{k-1}}_{v} & (12) \\ \hat{\mathbf{y}}^{C_k} &= \ominus \hat{\mathbf{x}}^{C_{k-1}}_{C_k} \oplus \hat{\mathbf{y}}^{C_{k-1}} . & (13) \end{aligned}$$

The final covariance is computed using the Jacobian of the composition equation $\mathbf{J}_{C_{k-1} \rightarrow C_k}$:

$$\mathbf{P}^{C_k}_{k} = \mathbf{J}_{C_{k-1} \rightarrow C_k} \mathbf{P}^{C_{k-1}}_{k} \mathbf{J}^{\top}_{C_{k-1} \rightarrow C_k} . \tag{14}$$

III. 1-Point RANSAC for EKF Estimation

While at least 5 points would be needed to compute monocular Structure from Motion for a calibrated camera if no prior knowledge is available [18], fewer are needed as more information is introduced: as few as 2 points in [19] for planar motion and 1 point in [25] for planar and nonholonomic motion. As a clear limitation of both approaches, any motion performed out of the model will result in estimation error. In fact, it is shown in real-image experiments in [25] that, although the most constrained model is enough for RANSAC hypotheses (reaching then 1-point RANSAC), a less restrictive one offers better results for motion estimation.

In the case of the new 1-point RANSAC presented here, extra information for the camera motion comes from the probability distribution function that the EKF naturally propagates over time. No camera motion constraints need to be added in order to successfully use a 1-point RANSAC hypothesis, and the full 6 degrees of freedom of camera motion can be dealt with using the proposed method.

Computational savings with respect to standard RANSAC can be easily derived. The number of RANSAC random hypothesis n_{hyp} necessary to guarantee that at least one of them is mismatch-free with probability p can be computed using the following formula:

$$n_{hyp} = \frac{log\left(1-p\right)}{log\left(1-\left(1-\epsilon\right)^{m}\right)} , \tag{15}$$

where ϵ is the assumed inlier ratio and m the number of measurements that instantiate the model. As a simple but illustrative example, if the inlier ratio ϵ is 0.5 and the probability p equals 0.99, the number of random hypothesis would be reduced from 146 ($m=5$; no prior information used) to only 7 for $m = 1$ (1-point RANSAC using prior information).

Efficiency in our 1-point RANSAC also derives from the fact that random RANSAC hypothesis evaluation does not imply an expensive EKF covariance update, whose complexity is cubic in the size of the estimated parameters. Support for each hypothesis is evaluated by comparing the innovation against a threshold; only a state update is necessary to compute it.

Our matching algorithm is fully detailed in pseudocode in Algorithm 1. It can be divided into two steps, which will be detailed in the following subsections.

A. First step: Generate a reliable set of low-innovation inliers

The input to the 1-point RANSAC algorithm is an initial set of individually compatible matches $\mathbf{z}$, extracted by cross-correlation search in the 99% probability region given by the predicted probability distribution function over the measurements $\mathcal{N}\left(\mathbf{h}_k\left(\hat{\mathbf{x}}_{k|k-1}\right), \mathbf{S}_k\right)$ [9]. Although this so-called active search filters out several outliers, the initial set $\mathbf{z}$ may still contain some of them.

The hypothesize-and-vote loop, where the main difference with plain RANSAC resides, operates as follows: first, a random match $\mathbf{z}^i$ is extracted from the individually compatible set $\mathbf{z}$. Using this single random match and the state prediction from the EKF (following the Gaussian $\mathcal{N}\left(\hat{\mathbf{x}}_{k|k-1}, \mathbf{P}_{k|k-1}\right)$) the state is updated. It is worth noticing again that only the state $\hat{\mathbf{x}}$ is updated in the loop and not the covariance $\mathbf{P}$, keeping the computational complexity low.

After each 1-match state update, the residual innovation is computed for the rest of the matches. Those below a heuristic threshold, in our experiments 1.0 pixels, are considered low-innovation inliers and support the current hypothesis.

Model hypothesis and verification is repeated as above and detailed in formula 15. Residual low-innovation inliers with the most supported hypothesis are definitely considered as inliers and the rest of the matches are either outliers or residual high-innovation inliers.

B. Second step: Rescue high-innovation inliers

As noted in the previous subsection, residual innovation will be high for outliers but also for some high-innovation inliers, which correspond to recently initialized points with uncertain depth estimates or close points significantly affected by camera translation. In order to finally discard spurious matches from this high-innovation set, the following rule is applied: high-innovation inliers will be individually compatible after a partial update eliminating correlated error; while outliers will not.

A state and covariance update is therefore carried out using all the low-innovation inliers from the first step. Individual compatibility will be evaluated for each one of high-innovation matches; those inside the 99% probability ellipse will be accepted as inliers, while those outside the ellipse will finally classified as outliers.

Algorithm 1 1-Point RANSAC for EKF estimation

```
INPUT: $\hat{\mathbf{x}}_{k|k-1}, \mathbf{P}_{k|k-1}$ {EKF prediction}
          $\mathbf{z}$ {Initial matches, may contain mismatches}
          $th$ {In this paper, $th = 1.0\ pixels$}
          $n\_hyp$
OUTPUT: $\mathbf{z}^{inliers}$ {A mismatch-free set of matches}

{1. Get a reliable set of low-innovation inliers}
$\mathbf{z}^{inliers} = [\ ]$
for $i = 0$ to $n\_hyp$ do
    $\mathbf{z}^i = select\_random\_match(\mathbf{z})$
    $\hat{\mathbf{x}}^i = EKF\_state\_update(\mathbf{z}^i, \hat{\mathbf{x}}_{k|k-1})$ {Notice: only state update; NO covariance update}
    $\mathbf{h}^i = predict\_all\_measurements(\hat{\mathbf{x}}^i)$
    $\mathbf{z}^i_{th} = find\_matches\_below\_a\_threshold(\mathbf{z}, \mathbf{h}^i, th)$
    if $size(\mathbf{z}^i_{th}) > size(\mathbf{z}^{inliers})$ then
        $\mathbf{z}^{inliers} = \mathbf{z}^i_{th}$ {Save the most supported hypothesis}
    end if
end for

{2. Rescue high-innovation inliers}
$[\hat{\mathbf{x}}, \mathbf{P}] = EKF\_update(\mathbf{z}^{inliers}, \hat{\mathbf{x}}_{k|k-1}, \mathbf{P}_{k|k-1})$
for every match $j$ above a threshold $th$ do
    $[\mathbf{h}^j, \mathbf{S}^j] = point\_j\_prediction\_and\_covariance(\hat{\mathbf{x}}, \mathbf{P}, j)$
    $\nu^{\mathbf{j}} = \mathbf{z}^j - \mathbf{h}^j$
    if $\nu^{\mathbf{j}\top}\mathbf{S}^{\mathbf{j}^{-1}}\nu^{\mathbf{j}} < \chi^2_{\mathbf{2},\mathbf{0.01}}$ then
        $\mathbf{z}^{inliers} = add\_match\_j\_to\_inliers(\mathbf{z}^{inliers}, \mathbf{z}^j)$ {If individually compatible, add to inliers}
    end if
end for
```

IV. Experimental results

In order to facilitate comparisons with the results obtained in this paper, the publicly available datasets *RAWSEEDS* [24] have been chosen to prove the validity of the proposed algorithm. These datasets have been captured with a multisensor platform both in indoor and outdoor environments. In this paper, three different outdoor sequences have been selected in order to perform a ground truth comparison against the Real Time Kinematics Differential GPS data provided in those datasets.

A. Ground truth comparison

Our goal is to compare EKF-based trajectory estimation against GPS measurements which we will consider as ground truth. As our EKF estimation takes the first camera frame C_0 as the origin of the frame of reference, a similarity transformation has to be applied that aligns every point of the trajectory $\mathbf{r}^{C_0}_{C_k} = \left[x^{C_0}_{C_k}\ y^{C_0}_{C_k}\ z^{C_0}_{C_k}\right]^\top$ with the ground truth GPS data $\mathbf{r}^{W}_{GPS_k}$, whose frame of reference we will denote by W:

$$\begin{bmatrix} \mathbf{r}^W_{C_k} \\ 1 \end{bmatrix} = \begin{bmatrix} x^W_{C_k} \\ y^W_{C_k} \\ z^W_{C_k} \\ 1 \end{bmatrix} = \begin{bmatrix} s\mathbf{R}^W_{C_0} & \mathbf{t}^W_{C_0} \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} x^{C_0}_{C_k} \\ y^{C_0}_{C_k} \\ z^{C_0}_{C_k} \\ 1 \end{bmatrix} . \quad (16)$$

In the above equation, $\mathbf{R}^W_{C_0}$ and $\mathbf{t}^W_{C_0}$ represent the rotation and translation between the GPS reference frame and the first camera reference frame, and s stands for the scale factor. The value of $\mathbf{t}^W_{C_0}$ can be taken from the GPS data in the first camera frame. $\mathbf{R}^W_{C_0}$ and s are unknown, and will be obtained via a non-linear optimization that minimizes the error between the aligned trajectory $\mathbf{r}^W_{C_k}$ and the ground truth $\mathbf{r}^W_{GPS_k}$.

For the sake of simplicity, the assumption that the position of the camera sensor and the GPS antenna are the same on the robot has been made in the above reasoning. As the position of the sensors in the robot differ by only a few centimetres, this assumption is reasonable in the experiments presented where the robot paths cover hundreds of metres and the error magnitudes are several metres.

Finally, the error of each camera position in the reconstructed path is computed as the Euclidean distance between each point of the estimated camera path and GPS path, both in the W reference,

$$e_k = \sqrt{\left(\mathbf{r}^W_{C_k} - \mathbf{r}^W_{GPS_k}\right)^\top \left(\mathbf{r}^W_{C_k} - \mathbf{r}^W_{GPS_k}\right)}. \quad (17)$$

B. Accuracy evaluation

Three different sequences from the *RAWSEEDS* dataset have been used to test the validity of the algorithm. All sequences were recorded with the same camera, a 320×240 resolution Unibrain camera with a wide-angle lens capturing at 30 fps. Camera calibration is provided in the dataset.

In the first sequence, consisting of 6000 images, the robot translates for about 146 metres. The second sequence has 5400 images and the robot describes a similar trajectory length, about 153 metres. Finally, a very large challenging sequence is evaluated that consists of 24180 frames (13.5 minutes of video) in which the robot describes a trajectory of 650 metres. In this latter sequence, although the accumulated drift makes the error noticeable when plotted with the GPS ground truth data, the relative error with respect to the trajectory keeps the same low value as the other two shorter sequences (1% of the trajectory length).

Figures 3, 4 and 5 show the estimated trajectory (in black) and the GPS ground truth (in red) over a top view extracted from Google Maps for each one of the sequences. From plain visual inspection, it can be seen in these figures that the estimated trajectory is not very far from the GPS trajectory. Table I details the maximum and mean errors obtained in the experiments. Figure 6 shows histograms of the errors for the three sequences.

V. Conclusions

A combination of sensor-centered EKF plus 1-point RANSAC has been presented in this paper, achieving sim-

Fig. 2. Example of images taken from the 650 metres monocular sequence.

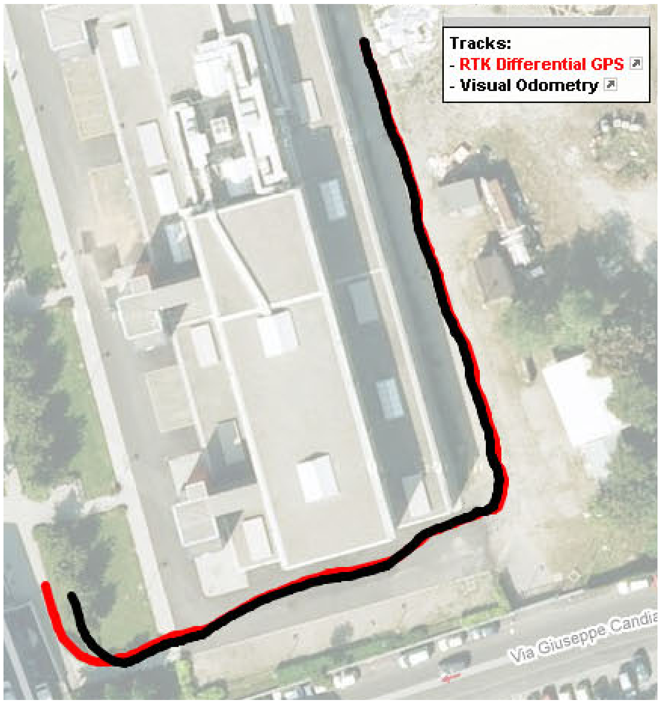

Fig. 3. Estimation results for the 146 metres trajectory superimposed on Google Maps. The trajectory estimated from vision is shown in red; GPS data is plotted in green. The mean error of the estimated trajectory is 1.3 metres.

TABLE I

EKF-based visual odometry error in the three experiments.

Trajectory length [m]	Mean error [m]	Maximum error [m]	% mean error over the trajectory
146	1.3	4.2	0.9%
153	1.9	3.3	1.1%
650	6.4	11.1	1.0%

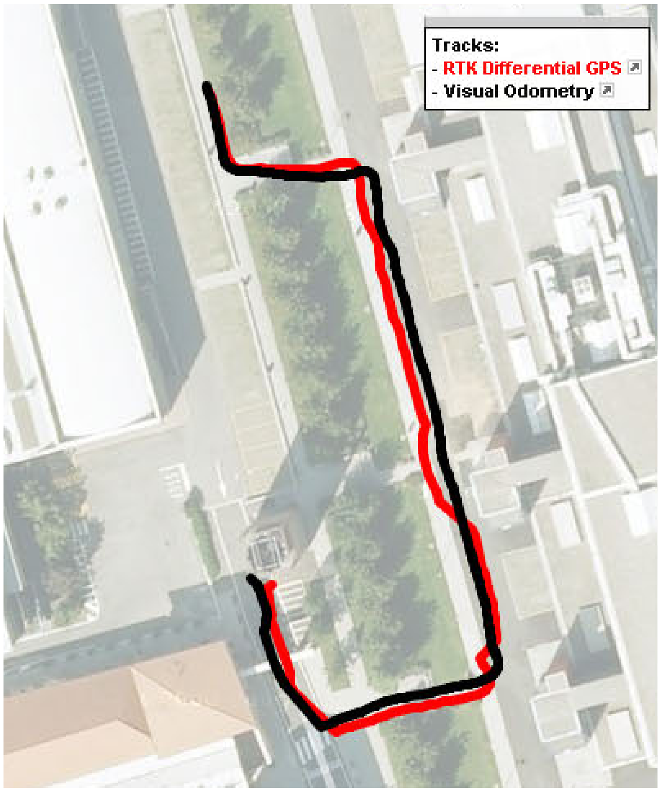

Fig. 4. Estimation results for the 153 metres trajectory. The mean error is 1.9 metres. Largest discrepancies between trajectories in this experiment, clearly visible for example in the right bottom corner, come from GPS error in the presence of high buildings.

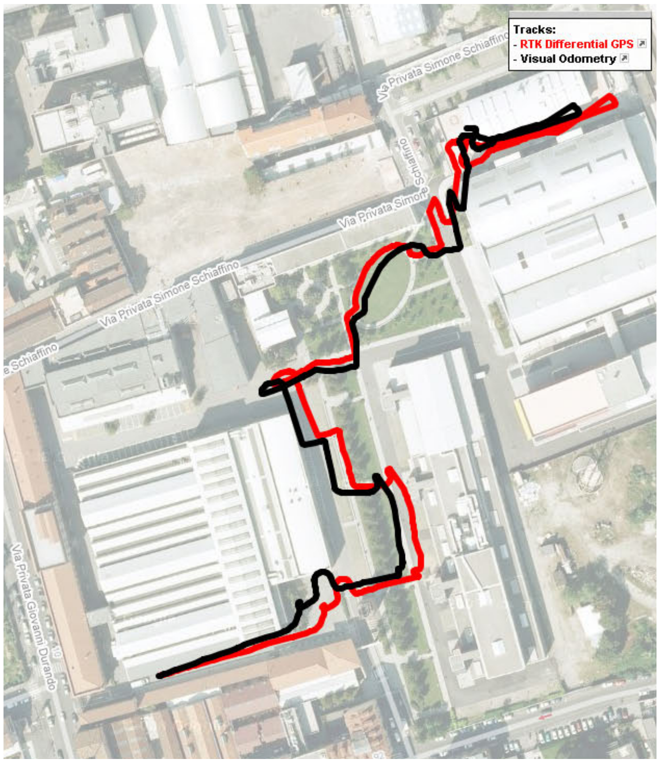
Tracks:
- RTK Differential GPS
- Visual Odometry
Via Privata Simone Schiaffino
Via Privata Simone Schiaffino
Via Privata Simone Schiaffino
Via Privata Simone
a Schiaffino
Via Privata Giovanni Durando

Fig. 5. Estimation results for the 650 metres trajectory. The mean error is 6.4 metres.

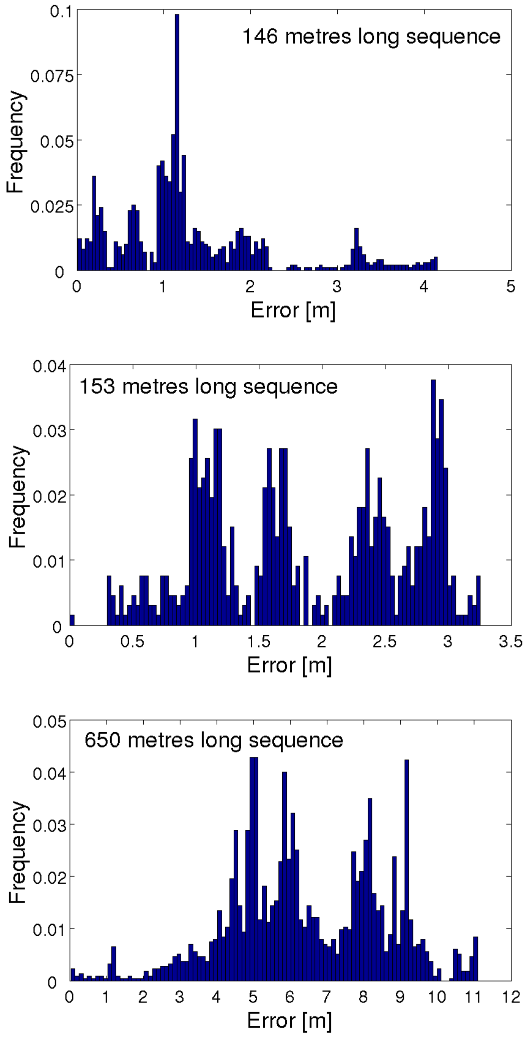

Fig. 6. Histograms of the errors for the three sequences.

ilar accuracy standards to state-of-the-art Visual Odometry algorithms based on non-linear optimization techniques.

Referring all the map features with respect to the current camera location highly reduces the linearization error, allowing us to deal with long trajectories. The new 1-point RANSAC algorithm offers interesting possibilities when compared with standard outlier rejection techniques, like traditional RANSAC and JCBB. Reducing the subset size to its minimum of 1 match, thanks to the probabilistic EKF prediction, allows us to greatly lower the number of random hypotheses and the computational cost with respect to Fischler and Bolles' RANSAC. With respect to JCBB, the random hypotheses that replace the exponential Branch and Bound search reduce the computational complexity of the algorithm, without losing the desirable quality of taking advantage of probabilistic prediction.

The proposed algorithm has been evaluated using a publicly available real-image dataset, using long sequences of thousands of frames taken by a robot in trajectories up to 650 metres. The comparison with GPS ground truth shows the accuracy of the method, which has a mean error of about 1% of the covered trajectory in the worst case. The three real-image experiments presented in this paper are the longest ever processed with a plain EKF without an additional technique like submapping or loop closing.

VI. Future Work

Although the 1-point RANSAC algorithm presented in the paper has experimentally proven itself efficient, a more thorough evaluation would be needed involving robustness and efficiency against plain RANSAC and JCBB. It would also be very interesting to compare ourselves with very recent mismatch rejection algorithms for filtering; particularly the already referenced Active Matching [5] and Randomized Joint Compatibility (RJC) [21]. The latter algorithm reduces the complexity of JCBB by ensuring an initial small set of jointly compatible inliers in a first step and checking afterwards joint compatibility of each remaining match and the initial small set.

It could also be of interest to explore and adapt to our algorithm recent RANSAC improvements commented on in the introduction, in search of a further reduction in the hypothesis generation and validation step.

Computational cost issues should also be evaluated in more detail. In the presented camera-centered EKF plus 1-point RANSAC the complexity is dominated by the cubic cost in the measurement vector size for the EKF update. The 1-point RANSAC represents a small fraction of the total cost, being suitable for real-time performance at 30 frames per second under similar conditions than [9], [7] (state vector size around 300 and measuring $10-20$ features per frame). The experiments of the paper measuring more than a hundred features are currently running at 1 Hz in an Intel Core 2 Quad at 2.83 GHz.

It is interesting to notice that, although this paper is focused on the particular case of EKF visual estimation, the new 1-point RANSAC presented here is independent of the type of sensor used. The only requirement of the algorithm is the availability of highly correlated prior information, typical of EKF SLAM for any kind of sensor used —and also in the multisensor case. Also, as highly correlated priors are not exclusive of EKF SLAM, the applicability of the 1-point RANSAC could be even broader. As an example, we think that camera pose tracking in keyframe schemes would benefit from our 1-point RANSAC cost reduction if a dynamic model were added to predict camera motion between frames.

Acknowledgments

This work was supported by projects DPI2006-13578 and DPI2009-07130 (Dirección General de Investigación of Spain), RAWSEEDS FP6-045144 and European Research Council Starting Grant 210346. The authors are grateful to Brian Williams, Ian Reid, Frank Dellaert and J. D. Tardós for fruitful discussions; and members of A. J. Davison's group (A. Angeli, M. Chli, G. Carrera, S. Lovegrove, H. Strasdat, A. Handa, P. Mountney and R. Newcombe) for software collaboration.

References

[1] A. Azarbayejani and A. P. Pentland. Recursive estimation of motion, structure, and focal length. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(6):562–575, June 1995.

[2] D. Capel. An effective bail-out test for ransac consensus scoring. In *Proceedings of the British Machine Vision Conference*, 2005.

[3] J. Castellanos, J. Neira, and J. Tardos. Limits to the consistency of EKF-based SLAM. In *5th IFAC Symposium on Intelligent Autonomous Vehicles*, 2004.

[4] A. Chiuso, P. Favaro, H. Jin, and S. Soatto. "MFm": 3-D motion from 2-D motion causally integrated over time. In *European Conference on Computer Vision*, pages 735–750, 2000.

[5] M. Chli and A. Davison. Active Matching. In *Proceedings of the 10th European Conference on Computer Vision: Part I*, pages 72–85, 2008.

[6] O. Chum and J. Matas. Optimal randomized RANSAC. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2008.

[7] J. Civera, A. J. Davison, and J. M. M. Montiel. Inverse depth parametrization for monocular SLAM. *IEEE Transactions on Robotics*, 24(5):932–945, October 2008.

[8] L. A. Clemente, A. J. Davison, I. D. Reid, J. Neira, and J. D. Tardos. Mapping large loops with a single hand-held camera. In *Proceedings of Robotics: Science and Systems*, 2007.

[9] A. J. Davison, N. D. Molton, I. D. Reid, and O. Stasse. MonoSLAM: Real-time single camera SLAM. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, June:1052–1067, 2007.

[10] E. Eade and T. Drummond. Monocular slam as a graph of coalesced observations. In *IEEE 11th International Conference on Computer Vision*, pages 1–8, 2007.

[11] M. A. Fischler and R. C. Bolles. Random sample consensus, a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6), 1981.

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

[13] G. Klein and D. Murray. Parallel tracking and mapping for small AR workspaces. In *Proc. Sixth IEEE and ACM International Symposium on Mixed and Augmented Reality*, 2007.

[14] K. Konolige and M. Agrawal. FrameSLAM: From bundle adjustment to realtime visual mappping. *IEEE Transactions on Robotics*, 24(5):1066–1077, 2008.

[15] F. Moreno-Noguer, V. Lepetit, and P. Fua. Pose Priors for Simultaneously Solving Alignment and Correspondence. In *10th European Conference on Computer Vision*, 2008.

[16] E. Mouragnon, M. Lhuillier, M. Dhome, F. Dekeyser, and P. Sayd. Real-time localization and 3D reconstruction. In *Proceedings of the IEEE Computer Vision and Pattern Recognition*, volume 1, 2006.

[17] J. Neira and J. D. Tardós. Data association in stochastic mapping using the joint compatibility test. *IEEE Transactions on Robotics and Automation*, 17(6):890–897, 2001.

[18] D. Nistér. An efficient solution to the five-point relative pose problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(6):756–770, 2004.

[19] D. Ortín and J. M. M. Montiel. Indoor robot motion based on monocular images. *Robotica*, 19(03):331–342, 2001.

[20] L. Paz, P. Piniés, J. Tardós, and J. Neira. Large-Scale 6-DOF SLAM With Stereo-in-Hand. *IEEE Transactions on Robotics*, 24(5), 2008.

[21] L. Paz, J. Tardos, and J. Neira. Divide and Conquer: EKF SLAM in O (n). *IEEE Transactions on Robotics*, 24(5):1107–1120, 2008.

[22] P. Piniés and J. Tardós. Large Scale SLAM Building Conditionally Independent Local Maps: Application to Monocular Vision. *IEEE Transactions on Robotics*, 24(5), 2008.

[23] R. Raguram, J. Frahm, and M. Pollefeys. A Comparative Analysis of RANSAC Techniques Leading to Adaptive Real-Time Random Sample Consensus. In *European Conference on Computer Vision*, 2008.

[24] RAWSEEDS. *RAWSEEDS* public datasets web page. URL http://www.rawseeds.org/, June 2009.

[25] D. Scaramuzza, F. Fraundorfer, and R. Siegwart. Real-Time Monocular Visual Odometry for On-Road Vehicles with 1-Point RANSAC. In *EEE International Conference on Robotics and Automation*.

[26] G. Sibley, C. Mei, I. Reid, and P. Newman. Adaptive relative bundle adjustment. In *Proceedings of Robotics: Science and Systems*, pages 976–982, Seattle, USA, June 2009.

[27] B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon. Bundle adjustment – A modern synthesis. In *Vision Algorithms: Theory and Practice*, LNCS, pages 298–375. Springer Verlag, 2000.

[28] A. Vedaldi, H. Jin, P. Favaro, and S. Soatto. KALMANSAC: Robust filtering by consensus. In *Tenth IEEE International Conference on Computer Vision, 2005. ICCV 2005*, volume 1, 2005.