1-Point RANSAC for EKF Filtering. Application to Real-Time Structure from Motion and Visual Odometry

Javier Civera
Robotics, Perception and Real-Time Group
Universidad de Zaragoza, Spain
jcivera@unizar.es

Oscar G. Grasa
Robotics, Perception and Real-Time Group
Universidad de Zaragoza, Spain
oscgg@unizar.es

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

J. M. M. Montiel
Robotics, Perception and Real-Time Group
Universidad de Zaragoza, Spain
josemari@unizar.es

Abstract

Random Sample Consensus (RANSAC) has become one of the most successful techniques for robust estimation from a data set that may contain outliers. It works by constructing model hypotheses from random minimal data subsets and evaluating their validity from the support of the whole data. In this paper we present a novel combination of RANSAC plus Extended Kalman Filter (EKF) that uses the available prior probabilistic information from the EKF in the RANSAC model hypothesize stage. This allows the minimal sample size to be reduced to one, resulting in large computational savings without the loss of discriminative power. 1-Point RANSAC is shown to outperform both in accuracy and computational cost the Joint Compatibility Branch and Bound (JCBB) algorithm, a gold-standard technique for spurious rejection within the EKF framework.

Two visual estimation scenarios are used in the experiments: First, six degrees of freedom motion estimation from a monocular sequence (Structure from Motion). Here, a new method for benchmarking six DOF visual estimation algorithms based on the use of high resolution images is presented, validated and used to show the superiority of 1-Point RANSAC. Second, we demonstrate long-term robot trajectory estimation combining monocular vision and wheel odometry (Visual Odometry). Here, a comparison against GPS shows an accuracy comparable to state-of-the-art visual odometry methods.

1 Introduction

The establishment of reliable correspondences from sensor data is at the core of most estimation algorithms in robotics. The search for correspondences, or data association, is usually based first stage on comparing local descriptors of salient features in the measured data. The ambiguity of such local description leads to possible incorrect correspondences at this stage. Robust methods operate by checking the consistency of the data against the global model assumed to be generating the data, and discarding as spurious any that does not fit into it. Among robust estimation methods, Random Sample Consensus (RANSAC) (Fischler and

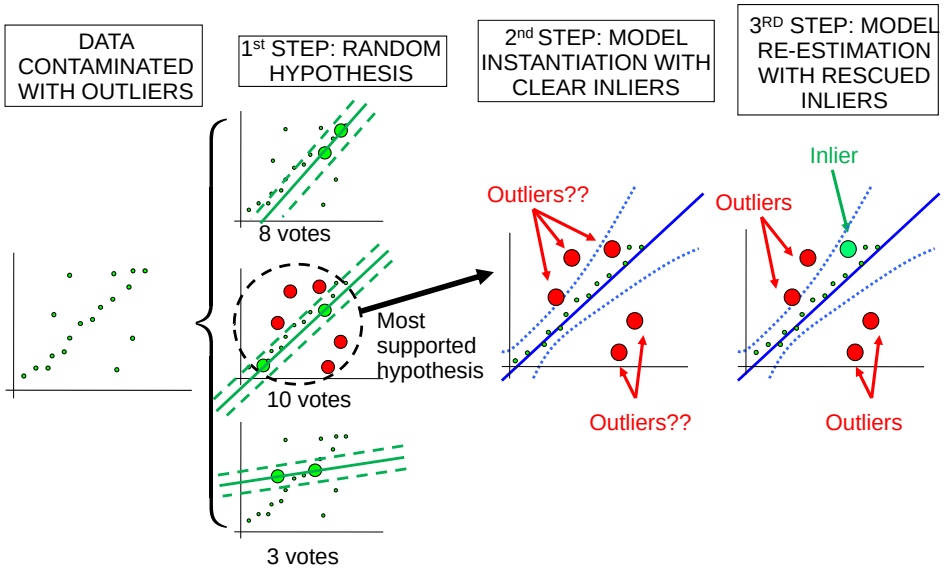


Figure 1: RANSAC steps for the simple 2D line estimation example: First, random hypotheses are generated from data samples of size two, the minimum to define a line. The most supported one is selected, and data voting for this hypothesis is considered inlier. Model parameters are estimated from those clear inliers in a second step. Finally, the remaining data points consistent with this latest model are rescued and the model is re-estimated again.

Bolles, 1981) stands out as one of the most successful and widely used, especially in the Computer Vision community.

This paper introduces a novel integration of RANSAC into the Extended Kalman Filter framework. In order to highlight the requirements and benefits of our method, the RANSAC algorithm is first briefly exposed in this introduction for the simple case of 2D line estimation from a set of points contaminated with spurious data (see Figure 1). After that, the same simple example will be tackled using the proposed 1-Point RANSAC algorithm (Figure 2). It is important to remark here that we use this simple example only to illustrate in the simplest manner our approach, and will later on fill in the details which make 1-Point RANSAC into a fully practical matching algorithm.

Standard RANSAC starts from a set of data, in our simple example 2D points, and the underlying model that generates the data, a 2D line. In the first step, RANSAC constructs hypotheses for the model parameters and selects the one that gathers most support. Hypotheses are randomly generated from the minimum number of points necessary to compute the model parameters, which is two in our case of line estimation. Support for each hypothesis can be computed in its most simple form by counting the data points inside a threshold (related to the data noise), although more sophisticated methods have been used (Torr and Zisserman, 2000).

Hypotheses involving one or more outliers are assumed to receive low support, as is the case in the third hypothesis in Figure 1. The number of hypotheses n_{hyp} necessary to ensure that at least one spurious-free hypothesis has been tested with probability p can be computed from this formula:

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

where ϵ is the outlier ratio and m the minimum number of data points necessary to instantiate the model. The usual approach is to adaptively compute this number of hypotheses at each iteration, assuming the inlier ratio is the support set by the total number of data points in this iteration (Hartley and Zisserman,

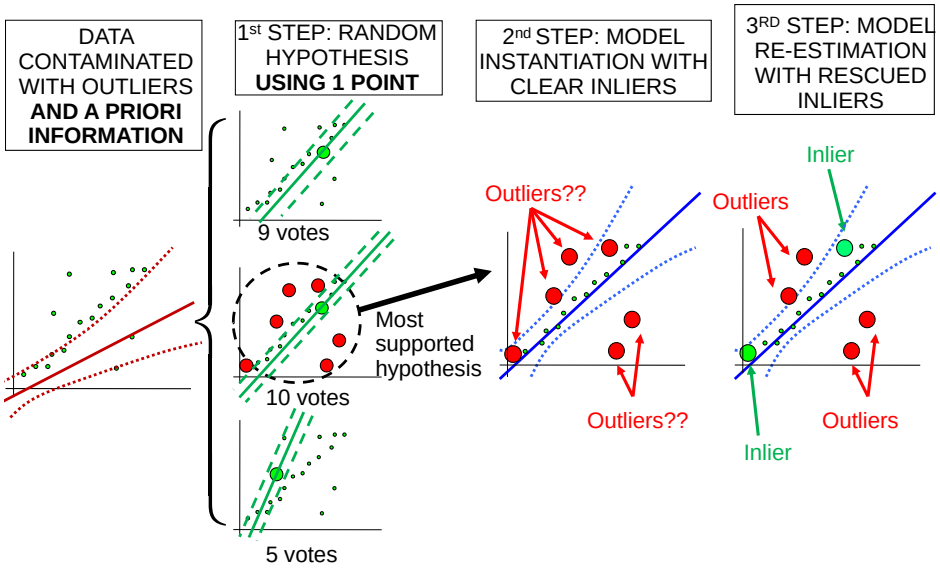


Figure 2: 1-Point RANSAC steps for the simple 2D line estimation example: As a key difference from standard RANSAC, the algorithm assumes that an a priori probability distribution over the model parameters is known in advance. This prior knowledge allows us to compute the random hypotheses using only 1 data point, hence reducing the number of hypotheses and the computational cost. The remaining steps do not vary with respect to standard RANSAC in Figure 1

.

2004).

Data points that voted for the most supported hypothesis are considered clear inliers. In a second stage, clear inliers are used to estimate the model parameters. Individual compatibility is checked for each one of the rest of the points against the estimated model. If any of them is rescued as inlier, as happens in the example in Figure 1, the model parameters are re-estimated again in a third step.

Figure 2 illustrates the idea behind 1-Point RANSAC in the same 2D line estimation problem. As the first key difference, the starting point is a data set and its underlying model, but also a prior probability distribution over the model parameters. RANSAC random hypotheses are then generated based on this prior information and data points, differently from standard RANSAC hypothesis solely based on data points. The use of prior information can reduce the size of the data set that instantiates the model to the minimum size of one point, and it is here where the computational benefit of our method with respect to RANSAC arises: according to Equation 1, reducing the sample size m greatly reduces the number of RANSAC iterations and hence the computational cost.

The order of magnitude of this reduction can be better understood if we switch from the simple 2D line estimation example to our visual estimation application. According to (Nistér, 2004), at least five image points are necessary to estimate the 6 degrees of freedom camera motion between two frames (so $m = 5$). Using formula 1, assuming an inlier ratio of 0.5 and a probability p of 0.99, the number of random hypotheses would be 146. Using our 1-Point RANSAC scheme, assuming that probabilistic a priori information is available, the sample size m can be reduced to one point and the number of hypotheses would be reduced to 7.

Having an a priori probability distribution over the camera parameters is unusual in classical pairwise Structure from Motion which assumes widely separated views (Hartley and Zisserman, 2004), and methods like standard RANSAC which generate hypotheses from candidate feature matches are mandatory in this case. But in sequential SfM from video (such as (Davison, 2003; Klein and Murray, 2008; Mouragnon et al.,

2009)), smooth interframe camera motion can be reasonably assumed and this used to generated a prior distribution (prediction) of the motion. For the specific EKF implementation of sequential SfM used in this paper, this prior probability is naturally propagated by the filter and is straightforwardly available.

The rest of the paper is organised as follows: first, related work is described in Section 2; the proposed algorithm is then described in its most general form in Section 3 and the details for the visual application are given in Section 4. Experimental results are shown in Section 5, including pure visual estimation and the monocular and wheel odometry combination. Finally, discussion, conclusions and future work are presented in sections 6, 7 and 8.

This paper builds on previous work in (Civera et al., 2009). The specific contributions of this journal version are: First, to extend the 1-point RANSAC initial raw idea from (Civera et al., 2009) to a fully detailed algorithm in which 1-point RANSAC is efficiently embedded in the standard EKF. Second, the proposal of a benchmarking method for 6 DOF camera motion estimation. And third, an extensive experimental validation of the algorithm. With more detail, the new experimental results benchmark our proposal against 5-point RANSAC and JCBB both in accuracy and cost using the proposed benchmarking method. Also, a real-time 1.3 kilometres long visual odometry experiment combining wheel odometry and camera information has been added.

2 Related Work

2.1 Random Sample Consensus (RANSAC)

Although RANSAC is a relatively old method, the literature covering the topic continues up to the present. RANSAC (Fischler and Bolles, 1981) was introduced early in visual geometric estimation (Torr and Murray, 1993) and has been the preferred outlier rejection tool in the field. Recently, an important stream of research has focused on reducing the model verification cost in standard RANSAC (e.g. (Raguram et al., 2008; Chum and Matas, 2008; Capel, 2005; Nistér, 2005)) via the early detection and termination of bad hypotheses. The 1-point RANSAC algorithm proposed here is related to this stream in the sense that it also reduces the hypothesis generation and validation cost. Nevertheless, it does so in a different manner: instead of fast identification of good hypotheses among a large number of them, the number of hypotheses is greatly reduced.

Incorporating probabilistic information into RANSAC has rarely been discussed in the computer vision literature. Only very recently Moreno *et al.* (Moreno-Noguer et al., 2008) have explored the case where weak a priori information is available in the form of probabilistic distribution functions.

More related to our research, the combination of RANSAC and Kalman filtering was proposed by Vedaldi *et al.* in (Vedaldi et al., 2005). Our method might be considered a specific form of Vedaldi's quite general approach. They propose an iterative scheme in which several minimal hypotheses are tested; for each such hypothesis all the consistent matches are iteratively harvested. No statement about the cardinality of the hypotheses is made. Here we propose a definite and efficient method, in which the cardinality of the hypotheses generator size is 1, and the inlier harvesting is not iterative but in two stages. Finally we describe in reproducible detail how to deal efficiently with the EKF algorithm in order to reach real-time, splitting the expensive EKF covariance update in two stages.

RANSAC using 1-point hypotheses has also been very recently proposed in (Scaramuzza et al., 2009) as the result of constraining the camera motion. While at least 5 points would be needed to compute monocular Structure from Motion for a calibrated camera undergoing general six degrees of freedom motion (Nistér, 2004), fewer are needed if the motion is known to be less general: as few as 2 points in (Ortín and Montiel, 2001) for planar motion and 1 point in (Scaramuzza et al., 2009) 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 (Scaramuzza et al., 2009) that although the most constrained model is enough for RANSAC hypotheses (reaching then 1-point RANSAC), a less restrictive model offers better results for motion estimation.

In the case of the new 1-point RANSAC presented here, extra information for the predicted camera motion comes from the probability distribution function that the EKF naturally propagates over time. The method presented is then in principle not restricted to any specific motion, being suitable for 6 degrees of freedom estimation. The only assumption is the existence of tight and highly correlated priors, which is reasonable within the EKF framework since the filter itself only works in such circumstances.

2.2 Joint Compatibility Branch and Bound (JCBB)

Joint Compatibility Branch and Bound (JCBB) (Neira and Tardós, 2001) has been the preferred technique for spurious match rejection within the EKF framework in the robotics community, being successfully used in visual (e.g. (Clemente et al., 2007), (Williams et al., 2007)) and non-visual SLAM (e.g. (Fenwick et al., 2002)). Unlike RANSAC, which hypothesizes model parameters based on current measurement data, JCBB detects spurious measurements based on a predicted probability distribution over the measurements. It does so by extracting from all the possible matches the maximum set that is jointly compatible with the multivariate Gaussian prediction.

In spite of its wide use, JCBB presents two main limitations that 1-Point RANSAC overcomes. First, JCBB operates over the prediction for the measurements *before* fusing them. Such a probabilistic prediction is coming from the linearization of the dynamic and measurement models and the assumption of Gaussian noise; so it will presumably not correspond to the real state of the system. 1-Point and in general any RANSAC operates over hypotheses *after* the integration of a data subset, which have corrected part of the predicted model error with respect to the real system.

The second limitation of JCBB concerns computational cost: the Branch and Bound search that JCBB uses for extracting the largest jointly compatible set of matches has exponential complexity in the number of matches. This complexity does not present a problem for small numbers of matches, as is the case in the references two paragraphs above, but very large computation times arise when the number of spurious grows, as we will show in the experimental results section. The computational complexity of 1-Point RANSAC is linear in the state and measurement size and exhibits low cost variation with the number of outliers.

Two recent methods are also of interest for this work. First, Active Matching (Chli and Davison, 2008) is a clear inspiration for our method. In Active Matching, feature measurements are integrated sequentially, with the choice of measurement at each step driven by expected information gain, and the results of each measurement in turn used to narrow the search for subsequent correspondences. 1-Point RANSAC can be seen as lying in the middle ground between RANSAC and JCBB which obtain point correspondence candidates and then aim to resolve them, and Active Matching with its fully sequential search for correspondence. The first step of 1-Point RANSAC is very similar to Active Matching, and confirming that integrating the first match highly constrains the possible image locations of other features, but afterwards the methods of the algorithms diverge. A problem with Active Matching in (Chli and Davison, 2008) was the unreasonably high computational cost of scaling to large numbers of feature correspondences per frame, and 1-Point RANSAC has much better properties in this regard, though very recently an improvement to Active Matching has also addressed this issue in a different way (Handa et al., 2010).

Paz *et al.* (Paz et al., 2008) describe can approach called Randomized Joint Compatibility (RJC) which basically randomizes the jointly compatible set search, avoiding the Branch and Bound search and ensuring an initial small set of jointly compatible inliers at the first step via Branch and Bound search in random sets. Only afterwards, the joint compatibility of each remaining match is checked against the initial set. Although this approach lowers the computational cost of the JCBB, it still faces the accuracy problems derived from the use of the predicted measurement function before data fusion.

2.3 Structure from Motion and Visual Odometry

Structure from Motion (SfM) is the generic term for 3D estimation from the sole input of a set of images of the imaged 3D scene and the corresponding camera locations. SfM from a sparse set of images has been usually processed by pairwise geometry algorithms (Hartley and Zisserman, 2004) and refined by global optimization procedures (Triggs et al., 2000). Estimation from a sequence has been carried out either by local optimization of keyframes (Klein and Murray, 2008; Mouragnon et al., 2009), or by filtering (Davison et al., 2007; Eade and Drummond, 2007). In our work we apply 1-Point RANSAC to filtering based SLAM using the EKF inverse depth parametrization of (Civera et al., 2008).

Visual Odometry, a term coined in (Nistér et al., 2004), refers to egomotion estimation mainly from visual input (monocular or stereo), but sometimes also combined with mechanical odometry and/or inertial sensor measurements. The variety of approaches here makes a complete review difficult; some visual odometry algorithms have made use of stereo cameras, either as the only sensor (e.g. (Comport et al., 2007)) or in combination with inertial measurements (Konolige et al., 2007; Cheng et al., 2006). Among the monocular approaches, (Mouragnon et al., 2009) uses a non-panoramic camera as the only sensor. Several others have been proposed using an omnidirectional camera, e.g. (Scaramuzza et al., 2009; Tardif et al., 2008). The experiment presented here, combining a non-panoramic camera plus proprioceptive information for estimation of large trajectories, is rarely found in the literature.

2.4 Benchmarking

Carefully designed benchmark datasets and methods have come into standard use in the vision community, e.g. (Scharstein and Szeliski, 2002). Robotics datasets have only recently reached such level of detail, presenting either detailed benchmarking procedures (Kummerle et al., 2009) or datasets with reliable ground truth and open resources for comparison (Smith et al., 2009; Blanco et al., 2009).

The RAWSEEDS dataset (RAWSEEDS, 2010), which include monocular and wheel odometry streams for large scale scenarios, will be used for the Visual Odometry experiments of the paper. While being suitable to benchmark very large real-image experiments, robotic datasets face two main inconveniences: First, the robot motion is planar in all the datasets, thus not allowing to evaluate full six-degrees-of-freedom motion estimation. And second, GPS only provides translational data and angular estimation cannot be benchmarked. Simulation environments, like the one described in (Funke and Pietzsch, 2009), can provide the translational and angular ground truth for any kind of camera motion. Nevertheless, those simulation environments usually cannot represent full real world complexity.

The benchmarking method proposed and used in the paper overcomes all these limitations. It consists of comparing the estimation results against a Bundle Adjustment solution over high resolution images. Full 6 DOF motion can be evaluated with low user effort (only the generation of a Bundle Adjustment solution is required), requirements for hardware are low (a high resolution camera) and any kind of motion or scene can be evaluated as the method operates over the real images themselves.

This approach is not entirely new: the use of a global Bundle Adjustment solution to benchmark sequential algorithms has already been used in (Eade and Drummond, 2007; Mouragnon et al., 2009). The contribution here is the validation of the algorithm, effectively showing that the Bundle Adjustment uncertainty is much lower than the sequential methods to benchmark. As another novelty, global Bundle Adjustment is applied over high resolution images, further improving accuracy. While it is true that a Bundle Adjustment solution still may suffer from scale drift, it will be much lower than that of the sequential algorithms. Also, scale drift can be driven close to zero by carefully chosing the images over which to apply Bundle Adjustment to form a well-conditioned network (Triggs et al., 2000), so the validity of the method is not compromised.

3 1-Point RANSAC Extended Kalman Filter Algorithm

Algorithm 1 outlines the proposed novel combination of 1-Point RANSAC inside the EKF framework in its most general form, and we describe this in detail in this section. The language used here is deliberately general in the belief that the described algorithm may be of application in a large number of estimation problems. The particular scenarios of the experimental results section (real-time sequential visual odometry from a monocular sequence, either with or without additional wheel odometry) are discussed in detail in section 4.

Algorithm 1 1-Point RANSAC EKF

INPUT: $\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{P}_{k-1|k-1}$ {EKF estimate at step $k-1$}
  th {Threshold for low-innovation points. In this paper, $th = 2\sigma\_pixels$}
OUTPUT: $\hat{\mathbf{x}}_{k|k}, \mathbf{P}_{k|k}$ {EKF estimate at step k}

{A. EKF prediction and individually compatible matches}
$[\hat{\mathbf{x}}_{k|k-1}, \mathbf{P}_{k|k-1}] = EKF\_prediction(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{P}_{k-1|k-1}, \mathbf{u})$
$[\hat{\mathbf{h}}_{k|k-1}, \mathbf{S}_{k|k-1}] = measurement\_prediction(\hat{\mathbf{x}}_{k|k-1}, \mathbf{P}_{k|k-1})$
$\mathbf{z}^{IC} = search\_IC\_matches(\hat{\mathbf{h}}_{k|k-1}, \mathbf{S}_{k|k-1})$

{B. 1-Point hypotheses generation and evaluation}
$\mathbf{z}^{li\_inliers} = [\ ]$
$n_{hyp} = 1000$ {Initial value, will be updated in the loop}
for $i = 0$ to n_{hyp} **do**
  $\mathbf{z}_i = select\_random\_match(\mathbf{z}^{IC})$
  $\hat{\mathbf{x}}_i = EKF\_state\_update(\mathbf{z}_i, \hat{\mathbf{x}}_{k|k-1})$ {Notice: only state update; NO covariance update}
  $\hat{\mathbf{h}}_i = predict\_all\_measurements(\hat{\mathbf{x}}_i)$
  $\mathbf{z}_i^{th} = find\_matches\_below\_a\_threshold(\mathbf{z}^{IC}, \hat{\mathbf{h}}_i, th)$
  if $size(\mathbf{z}_i^{th}) > size(\mathbf{z}^{li\_inliers})$ **then**
    $\mathbf{z}^{li\_inliers} = \mathbf{z}_i^{th}$
    $\epsilon = 1 - \frac{size(\mathbf{z}^{li\_inliers})}{size(\mathbf{z}^{IC})}$
    $n_{hyp} = \frac{log(1-p)}{log(1-(1-\epsilon))}$
  end if
end for

{C. Partial EKF update using low-innovation inliers}
$[\hat{\mathbf{x}}_{k|k}, \mathbf{P}_{k|k}] = EKF\_update(\mathbf{z}^{li\_inliers}, \hat{\mathbf{x}}_{k|k-1}, \mathbf{P}_{k|k-1})$

{D. Partial EKF update using high-innovation inliers}
$\mathbf{z}^{hi\_inliers} = [\ ]$
for every match $\mathbf{z}^j$ above a threshold th **do**
  $[\hat{\mathbf{h}}^j, \mathbf{S}^j] = point\_j\_prediction\_and\_covariance(\hat{\mathbf{x}}_{k|k}, \mathbf{P}_{k|k}, j)$
  $\boldsymbol{\nu}^{\mathbf{j}} = \mathbf{z}^j - \hat{\mathbf{h}}^j$
  if $\boldsymbol{\nu}^{\mathbf{j}\top} \mathbf{S}^{\mathbf{j}-\mathbf{1}} \boldsymbol{\nu}^{\mathbf{j}} < \chi^2_{\mathbf{2,0.01}}$ **then**
    $\mathbf{z}^{hi\_inliers} = add\_match\_j\_to\_inliers(\mathbf{z}^{hi\_inliers}, \mathbf{z}^j)$ {If individually compatible, add to inliers}
  end if
end for
if $size(\mathbf{z}^{hi\_inliers}) > 0$ **then**
  $[\hat{\mathbf{x}}_{k|k}, \mathbf{P}_{k|k}] = EKF\_update(\mathbf{z}^{hi\_inliers}, \hat{\mathbf{x}}_{k|k}, \mathbf{P}_{k|k})$
end if

3.1 EKF Prediction and Search for Individually Compatible Matches (lines 5–8)

The algorithm begins with standard EKF prediction: the estimation for the state vector $\mathbf{x}_{k-1|k-1}$ at step $k-1$, modeled as a multidimensional Gaussian $\mathbf{x}_{k-1|k-1} \sim \mathcal{N}\left(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{P}_{k-1|k-1}\right)$, is propagated to step k through the known dynamic model $\mathbf{f}_k$

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{f}_k\left(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{u}_k\right) \quad (2)$$

$$\mathbf{P}_{k|k-1} = \mathbf{F}_k \mathbf{P}_{k-1|k-1} \mathbf{F}_k^\top + \mathbf{G}_k \mathbf{Q}_k \mathbf{G}_k^\top \ . \quad (3)$$

In the above equation $\mathbf{u}_k$ stands for the control inputs to the system at step k, $\mathbf{F}_k$ is the Jacobian of $\mathbf{f}_k$ with respect to the state vector $\mathbf{x}_{k|k-1}$ at step k, $\mathbf{Q}_k$ is the covariance of the zero-mean Gaussian noise assumed for the dynamic model and $\mathbf{G}_k$ is the Jacobian of this noise with respect to the state vector $\mathbf{x}_{k|k-1}$ at step k.

The predicted probability distribution for the state $\mathbf{x}_{k|k-1}$ can be used to ease the correspondence search process by Active Search (Davison et al., 2007). Propagating this predicted state through the measurement model $\mathbf{h}_i$ offers a Gaussian prediction for each measurement:

$$\hat{\mathbf{h}}_i = \mathbf{h}_i\left(\hat{\mathbf{x}}_{k|k-1}\right) \quad (4)$$

$$\mathbf{S}_i = \mathbf{H}_i \mathbf{P}_{k|k-1} \mathbf{H}_i^\top + \mathbf{R}_i \ , \quad (5)$$

where $\mathbf{H}_i$ is the Jacobian of the measurement $\mathbf{h}_i$ with respect to the state vector $\mathbf{x}_{k|k-1}$ and $\mathbf{R}_i$ is the covariance of the Gaussian noise assumed for each individual measurement. The actual measurement $\mathbf{z}_i$ should be exhaustively searched for inside the 99% probability region defined by its predicted Gaussian $\mathcal{N}\left(\hat{\mathbf{h}}_i, \mathbf{S}_i\right)$ by comparison of the chosen local feature descriptor.

Active Search allows computational savings and also constraints the matches to be individually compatible with the predicted state $\mathbf{x}_{k|k-1}$. Nevertheless, ensuring geometric compatibility for each separated match $\mathbf{z}_i$ does not guarantee the global consensus of the whole set. So, still the joint compatibility of the data against a global model has to be checked for the set individually compatible matches $\mathbf{z}^{IC} = \left(\mathbf{z}_1 \dots \mathbf{z}_i \dots \mathbf{z}_n\right)^\top$ previous to the EKF update.

3.2 1-Point Hypotheses Generation and Evaluation (lines 9–22)

Following the principles of RANSAC, random state hypotheses $\hat{\mathbf{x}}_i$ are generated and data support is computed by counting measurements inside a threshold. It is assumed that we are considering problems where the predicted measurements are highly correlated, such that every hypothesis computed from one match reduces most of the correlation in the measurement prediction, with inlier uncertainty close to the measurement noise. The threshold is fixed according to a χ^2 test with significance $\alpha = 0.05$.

As the key difference with respect to standard RANSAC, random hypotheses will be generated not only based on the data $\mathbf{z}^{IC} = \left(\mathbf{z}_1 \dots \mathbf{z}_i \dots \mathbf{z}_n\right)^\top$ but also on the predicted state $\mathbf{x}_{k|k-1} \sim \mathcal{N}\left(\hat{\mathbf{x}}_{k|k-1}, \mathbf{P}_{k|k-1}\right)$. Exploiting this prior knowledge allows us to reduce the sample size necessary to instantiate the model parameters from the minimal size to define the degrees of freedom of the model to only one data point. The termination criteria of the RANSAC algorithm, stated in Equation 1, grows exponentially with the sample size and means a great reduction in the number of hypotheses.

Another key aspect for the efficiency of the algorithm is that each hypothesis $\hat{\mathbf{x}}_i$ generation only needs an EKF state update using a single match $\mathbf{z}_i$. A covariance update, which is of quadratic complexity in the size of the state, is not needed and the cost per hypothesis will be low. Hypothesis support is calculated by projecting the updated state into the camera, which can also be performed at very low cost compared with other stages in the EKF algorithm.

3.3 Partial Update with Low-Innovation Inliers (lines 23–24)

Data points voting for the most supported hypothesis $\mathbf{z}^{li\_inliers}$ are designated as low-innovation inliers. They are assumed to be generated by the true model, as they are at a small distance from the most supported hypothesis. The rest of the points can be outliers but also inliers, even if they are far from the most supported hypothesis.

A simple example from visual estimation can illustrate this: it is well known that distant points are useful for estimating camera rotation, while close points are necessary to estimate translation (Civera et al., 2008). In the RANSAC hypotheses generation step, a distant feature would generate a highly accurate 1-point hypothesis for rotation, while translation would remain inaccurately estimated. Other distant points would in this case have low innovation and would vote for this hypothesis. But as translation is still inaccurately estimated, nearby points would presumably exhibit high innovation even if they are inliers.

So after having determined the most supported hypothesis and the other points that vote for it, some inliers still have to be "rescued" from the high-innovation set. Such inliers will be rescued after a partial state and covariance update using only the reliable set of low-innovation inliers:

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k \left(\mathbf{z}^{li\_inliers} - \mathbf{h}' \left(\hat{\mathbf{x}}_{k|k-1} \right) \right) \quad (6)$$

$$\mathbf{P}_{k|k} = \left(\mathbf{I} - \mathbf{K}_k \mathbf{H}_k \right) \mathbf{P}_{k|k-1} \quad (7)$$

$$\mathbf{K}_k = \mathbf{P}_{k|k-1} \mathbf{H}_k^\top \left(\mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^\top + \mathbf{R}_k \right)^{-1} . \quad (8)$$

$\mathbf{H}_k = \left(\mathbf{H}_1 \dots \mathbf{H}_i \dots \mathbf{H}_{n'} \right)^\top$ stands for the Jacobian of the measurement equation $\mathbf{h}' \left(\hat{\mathbf{x}}_{k|k-1} \right)$ that projects the low-innovation inliers into the sensor space. $\mathbf{R}_k$ is the covariance assigned to the sensor noise.

3.4 Partial Update with High-Innovation Inliers (lines 25–35)

After a partial update using low-innovation inliers, most of the correlated error in the EKF prediction is corrected and the covariance is greatly reduced. This high reduction will be exploited for the recovery of high-innovation inliers: as correlations have weakened, consensus for the set will not be necessary to compute and individual compatibility will suffice to discard inliers from outliers.

An individual Gaussian prediction $\mathbf{h}^j \sim \mathcal{N} \left(\hat{\mathbf{h}}^j, \mathbf{S}^j \right)$ will be computed for each high innovation for every match $\mathbf{z}^j$ by propagating the state after the first partial update $\mathbf{x}_{k|k}$ through the projection model. The match will be accepted as an inlier if it lies within the 99% probability region of the predicted Gaussian for the measurement.

After testing all the high-innovation measurements a second partial update will be performed with all the points classified as inliers $\mathbf{z}^{hi\_inliers}$, following the usual EKF equations.

It is worth remarking here that splitting the EKF update does not have a noticeable effect on the computational cost. If n is the state size and m the measurement vector size, and in the usual SLAM case

where the state is much bigger than the locally measured set $n >> m$, the main EKF cost is the covariance update which is $\mathcal{O}\left(mn^2\right)$. If the update is divided into two steps of measurement vector sizes m_1 and m_2 ($m = m_1 + m_2$), this covariance update cost stays almost the same. Some other minor costs grow, like the Jacobian computation which has to be done twice. But also some others are reduced, like the measurement covariance inversion which is $\mathcal{O}\left(m^3\right)$. Nevertheless, the effect of the latter two is negligible and for most EKF estimation cases the cost is dominated by the covariance update and remains approximately the same.

4 1-Point RANSAC Extended Kalman Filter from a Monocular Sequence Input

As previously stated, the proposed 1-point RANSAC and EKF combination will be evaluated in this paper for the particular case of visual estimation from a monocular camera. In this section, the general method detailed in Section 3 specializes to this specific application.

4.1 State Vector Definition

The state vector at step k is composed of a set of camera parameters $\mathbf{x}_{C_k}^W$ and map parameters $\mathbf{y}^W$. All of these are usually referred to a static reference frame W, although there are some advantages in referring them to the current camera frame C_k (Section 4.4 describes this latter approach);

$$\hat{\mathbf{x}}_k^W = \begin{pmatrix} \hat{\mathbf{x}}_{C_k}^W \\ \hat{\mathbf{y}}^W \end{pmatrix}; \quad \mathbf{P}_k^W = \begin{pmatrix} \mathbf{P}_{C_k}^W & \mathbf{P}_{C_k y}^W \\ \mathbf{P}_{y C_k}^W & \mathbf{P}_y^W \end{pmatrix} . \tag{9}$$

The estimated map $\mathbf{y}^W$ is composed of n point features $\mathbf{y}_i^W$; $\mathbf{y}^W = \left(\mathbf{y}_1^{W\top} \ \dots \ \mathbf{y}_n^{W\top}\right)^\top$ .Point features are parametrized in inverse depth coordinates $\mathbf{y}_{i,ID}^W = \left(X_i^W \ Y_i^W \ Z_i^W \ \theta_i^W \ \phi_i^W \ \rho_i\right)^\top$ and converted to Euclidean parametrization $\mathbf{y}_{i,E}^W = \left(X_i^W \ Y_i^W \ Z_i^W\right)^\top$ if and when the projection equation becomes linear enough, as described in (Civera et al., 2008). The inverse depth parametrization stores in its six parameters the 3D camera position when the feature was initialized $\left(X_i^W \ Y_i^W \ Z_i^W\right)^\top$, the azimuth-elevation pair $\left(\theta_i^W \ \phi_i^W\right)^\top$ encoding the unit ray pointing to the feature and its inverse depth along the ray ρ_i.

4.2 Dynamic Model

The dynamic model applied to the camera depends on the information available. For the case of pure visual estimation from a monocular sequence, a constant velocity model is sufficient for smooth hand-held motion (Davison et al., 2007). The camera state is then formed by position $\mathbf{r}_{C_k}^W$, orientation $\mathbf{q}_{C_k}^W$, and linear and angular velocities $\mathbf{v}^W$ and ω^{C_k}:

$$\mathbf{x}_{C_k}^W = \begin{pmatrix} \mathbf{r}_{C_k}^W \\ \mathbf{q}_{C_k}^W \\ \mathbf{v}^W \\ \omega^{C_k} \end{pmatrix} . \tag{10}$$

The constant velocity model $\mathbf{f}_v$ equations are as follows:

$$\mathbf{f}_v = \begin{pmatrix} \mathbf{r}^W_{C_{k+1}} \\ \mathbf{q}^W_{C_{k+1}} \\ \mathbf{v}^W_{C_{k+1}} \\ \omega^C_{C_{k+1}} \end{pmatrix} = \begin{pmatrix} \mathbf{r}^W_{C_k} + \left(\mathbf{v}^W_{C_k} + \mathbf{V}^W\right)\Delta t \\ \mathbf{q}^W_{C_k} \times \mathbf{q}\left(\left(\omega^C_{C_k} + \Omega^C\right)\Delta t\right) \\ \mathbf{v}^W_{C_k} + \mathbf{V}^W \\ \omega^C_{C_k} + \Omega^C \end{pmatrix} , \tag{11}$$

where $\mathbf{V}^W$ and Ω^C are zero-mean Gaussianly distributed velocity noise coming from an impulse of acceleration.

When other sensorial information apart from the monocular sequence is available, it should be incorporated as input to the dynamic model. In this paper, the combination of monocular vision plus wheel odometry is analyzed. In this case, the camera state only needs to contain position and orientation $\mathbf{x}^W_{C_k} = \begin{pmatrix} \mathbf{r}^W_{C_k} \\ \mathbf{q}^W_{C_k} \end{pmatrix}$. In this paper, the classical model for a differential drive robot (Borenstein et al., 1996) has been chosen to model its dynamics.

4.3 Measurement Model

The measurement model used in the experiments of the paper is a pinhole camera model plus a two parameters radial distortion (Civera et al., 2008). The camera is assumed to be calibrated in advance. Inverse depth and Euclidean points in the state vector are first transformed to the camera reference frame:

$$\mathbf{h}^{C_k}_{i,ID} = \mathbf{R}^{C_k}_W\left(\mathbf{q}^W_{C_k}\right)\left(\rho_i \begin{pmatrix} X^W_i \\ Y^W_i \\ Z^W_i \end{pmatrix} - \mathbf{r}^W_{C_k}\right) + \mathbf{m}\left(\theta^W_i, \phi^W_i\right) \tag{12}$$

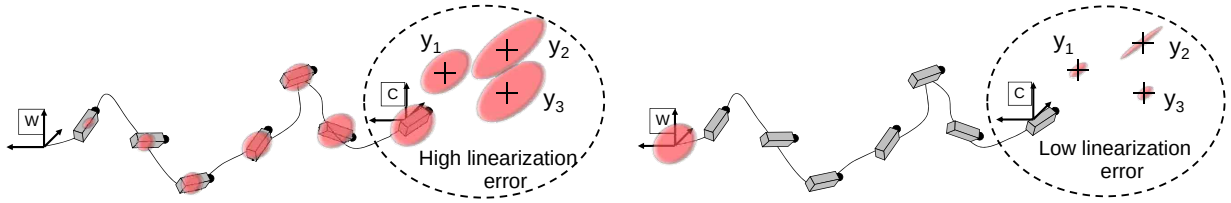
$$\mathbf{h}^{C_k}_{i,E} = \mathbf{R}^{C_k}_W\left(\mathbf{q}^W_{C_k}\right)\left(\mathbf{y}^W_{i,E} - \mathbf{r}^W_{C_k}\right) , \tag{13}$$

where $\mathbf{R}^{C_k}_W\left(\mathbf{q}^W_{C_k}\right)$ represents a rotation matrix computed from the state quaternion and $\mathbf{m}$ is the function converting azimuth-elevation angles to a unit vector. Points in the camera frame are then projected using the standard pinhole model:

$$\mathbf{h}_u = \begin{pmatrix} u_u \\ v_u \end{pmatrix} = \begin{pmatrix} u_0 - \frac{f}{d_x}\frac{h^C_x}{h^C_z} \\ v_0 - \frac{f}{d_y}\frac{h^C_y}{h^C_z} \end{pmatrix} . \tag{14}$$

Here f stands for the focal length of the camera and $(u_0, v_0)^\top$ are the image centre coordinates. The imaged point is finally transformed using the two parameter κ_1, κ_2 model below, resulting in the distorted measurement $\mathbf{h}_d = (u_d, v_d)^\top$

$$\begin{pmatrix} u_u \\ v_u \end{pmatrix} = \begin{pmatrix} u_0 + (u_d - u_0)\left(1 + \kappa_1 r_d^2 + \kappa_2 r_d^4\right) \\ v_0 + (v_d - v_0)\left(1 + \kappa_1 r_d^2 + \kappa_2 r_d^4\right) \end{pmatrix}$$
$$r_d = \sqrt{\left(d_x\left(u_d - u_0\right)\right)^2 + \left(d_y\left(v_d - v_0\right)\right)^2} . \tag{15}$$



(a) World-referenced EKF estimation. As the camera moves away from the World reference and does not revisit known places the uncertainty grows both in the camera and newly initialized features y_i. The wide uncertainty regions will produce high linearization errors and filter inconsistency.

(b) Camera-centered EKF estimation. The camera location uncertainty is close to zero, as the reference is always attached to it. Uncertainties for features in its vicinity y_i will be also kept low, so measurement model linearization errors will be kept small for the whole estimation.

Figure 3: Camera-centered and World-referenced EKF estimation.

4.4 Camera-Centered Estimation

It is well known that the usual EKF SLAM formulation, referred to a world reference frame, is only valid for local estimation in the surroundings of a sensor. Figure 3(a) illustrates the problem of this formulation: as the sensor moves away from the world reference, and if a pure exploratory trajectory is performed, the uncertainty of the estimation will always grow. Eventually it will reach a point where large linearization errors will cause inconsistency and filter divergence.

Figure 3(b) illustrates an alternative approach that alleviates this problem, that was first presented for EKF SLAM in (Castellanos et al., 2004). It basically consists of referring all geometric parameters to a reference frame attached to the camera. Uncertainty in the locality of the sensor will always be kept low, greatly reducing the linearization errors associated with the measurement model. The camera-centered approach was first used for visual EKF estimation in (Civera et al., 2009); and has been thoroughly benchmarked in (Williams, 2009).

The modifications with respect to world-centered visual SLAM are now given in detail. First, the state vector is composed of the location of the world reference frame $\mathbf{x}_W^{C_k}$ and the map of estimated features $\mathbf{y}^{C_k}$, both expressed in the current camera reference frame:

$$\mathbf{x}_k^{C_k} = \begin{pmatrix} \mathbf{x}_W^{C_k} \\ \mathbf{y}^{C_k} \end{pmatrix} . \tag{16}$$

The location of the world reference with respect to the current camera $\mathbf{x}_W^{C_k} = \begin{pmatrix} \mathbf{r}_W^{C_k} \\ \mathbf{q}_W^{C_k} \end{pmatrix}$ is coded with its position $\mathbf{r}_W^{C_k}$ and quaternion orientation $\mathbf{q}_W^{C_k}$. When odometry information is not available and a constant velocity model is assumed, velocities should also be included in the state $\mathbf{x}_k^{C_k} = \begin{pmatrix} \mathbf{x}_W^{C_k} \\ \mathbf{v}^{C_k} \\ \omega^{C_k} \\ \mathbf{y}^{C_k} \end{pmatrix}$.

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 $\mathbf{x}_{C_k}^{C_{k-1}}$ that represents the motion of the sensor between $k-1$ and k is added:

$$\mathbf{x}_{k|k-1}^{C_{k-1}} = \begin{pmatrix} \mathbf{x}_W^{C_{k-1}} \\ \mathbf{y}^{C_{k-1}} \\ \mathbf{x}_{C_k}^{C_{k-1}} \end{pmatrix} \tag{17}$$

The predicted camera motion is represented in terms of position and orientation, represented via a quaternion:

$$\mathbf{x}_{C_k}^{C_{k-1}} = \begin{pmatrix} \mathbf{r}_{C_k}^{C_{k-1}} \\ \mathbf{q}_{C_k}^{C_{k-1}} \end{pmatrix} . \tag{18}$$

The 1-point RANSAC EKF algorithm is applied with minor changes. The dynamic model of the system is applied over the motion relative to the previous frame contained in $\mathbf{x}_{C_k}^{C_{k-1}}$, either using the constant velocity model in equation 11 (in which case velocities should be kept then in the state as described above) or wheel odometry inputs. The measurement model described in Section 4.3 is modified only at its first step: as the map $\mathbf{y}^{C_{k-1}}$ is now in the previous camera frame C_{k-1}, Equations 12 and 13 change features from the previous to the current camera frame using relative motion in $\mathbf{x}_{C_k}^{C_{k-1}}$.

The algorithm proceeds then as explained in Section 3. At the end of the algorithm, after the second update, a rigid transformation is applied to change the reference frame from the previous camera to the current one. The world reference location is updated:

$$\mathbf{r}_W^{C_k} = \mathbf{R}_{C_{k-1}}^{C_k}\left(\mathbf{q}_{C_k}^{C_{k-1}}\right)\left(\mathbf{r}_W^{C_{k-1}} - \mathbf{r}_{C_k}^{C_{k-1}}\right) \tag{19}$$

$$\mathbf{q}_W^{C_k} = \mathbf{q}_W^{C_{k-1}} \times \mathbf{q}_{C_{k-1}}^{C_k} , \tag{20}$$

and the parameters representing motion from the previous to the current frame $\mathbf{x}_{C_k}^{C_{k-1}}$ are marginalized out from the state. Inverse depth and Euclidean map features are also affected by this composition step:

$$\mathbf{y}_{i,ID}^{C_k} = \begin{pmatrix} \mathbf{R}_{C_{k-1}}^{C_k}\left(\mathbf{q}_{C_k}^{C_{k-1}}\right)\begin{pmatrix} X_i^{C_{k-1}} \\ Y_i^{C_{k-1}} \\ Z_i^{C_{k-1}} \end{pmatrix} - \mathbf{r}_{C_k}^{C_{k-1}} \\ \mathbf{m}^{-1}\left(\mathbf{R}_{C_{k-1}}^{C_k}\left(\mathbf{q}_{C_k}^{C_{k-1}}\right)\mathbf{m}\left(\theta_i^{C_{k-1}}, \phi_i^{C_{k-1}}\right)\right) \\ \rho_i \end{pmatrix} ; \mathbf{y}_{i,E}^{C_k} = \mathbf{R}_{C_{k-1}}^{C_k}\left(\mathbf{q}_{C_k}^{C_{k-1}}\right)\left(\mathbf{y}_{i,E}^{C_{k-1}} - \mathbf{r}_{C_k}^{C_{k-1}}\right) . \tag{21}$$

The covariance is updated using the Jacobians of this composition function $\mathbf{J}_{C_{k-1}\rightarrow C_k}$

$$\mathbf{P}_k^{C_k} = \mathbf{J}_{C_{k-1}\rightarrow C_k}\mathbf{P}_k^{C_{k-1}}\mathbf{J}_{C_{k-1}\rightarrow C_k}^{\top} . \tag{22}$$

5 Experimental Results

5.1 Benchmark Method for 6 DOF Camera Motion Estimation.

The first step of the method takes an image sequence of the highest resolution, in order to achieve the highest accuracy. In this paper, a 1224×1026 pixels sequence was taken at 22 frames per second. A sparse

subset of n camera locations $\mathbf{x}_{BA}^{C_1}$ are estimated by Levenberg-Marquardt Bundle Adjustment with robust likelihood model (Triggs et al., 2000) over the correspongding n images in the sequence $\{I_1, \dots I_n\}$. Images are manually selected to ensure they form a strong network. The reference frame is attached to the camera C_1, corresponding to the first frame of the sequence I_1. For the experiments in the paper, 62 overlapping camera locations were reconstructed by manually matching 74 points spread over the images. $15-20$ points are visible in each image.

$$\mathbf{x}_{BA}^{C_1} = \begin{pmatrix} \mathbf{x}_{1,BA}^{C_1} \\ \vdots \\ \mathbf{x}_{n,BA}^{C_1} \end{pmatrix} , \tag{23}$$

$$\mathbf{x}_{i,BA}^{C_1} = \left(X_{i,BA}^{C_1} \, Y_{i,BA}^{C_1} \, Z_{i,BA}^{C_1} \, \phi_{i,BA}^{C_1} \, \theta_{i,BA}^{C_1} \, \psi_{i,BA}^{C_1} \right)^\top . \tag{24}$$

Each camera location is represented by its position $\left(X_{i,BA}^{C_1} \, Y_{i,BA}^{C_1} \, Z_{i,BA}^{C_1} \right)^\top$ and Euler angles $\left(\phi_{i,BA}^{C_1} \, \theta_{i,BA}^{C_1} \, \psi_{i,BA}^{C_1} \right)^\top$. The covariance of the solution is computed by back-propagation of reprojection errors $P_{BA}^{C_1} = \left(J^\top R^{-1} J \right)^{-1}$, where J is the Jacobian of the projection model and R is the covariance of the Gaussian noise assumed in the model.

The input sequence is then reduced by dividing its width and height by four. The algorithm to benchmark is applied over the subsampled sequence. The reference frame is also attached to the first camera C_1, which is taken to be the same first one as in Bundle Adjustment. Images for which a Bundle Adjustment estimation is available are selected and stored $\mathbf{x}_{i,MS}^{C_1}$, each along with its individual covariance $P_{i,MS}^{C_1}$ directly extracted from the EKF at each step.

As the reference has been set to the same first image of the sequence, the Bundle Adjustment and sequential estimation solutions only differ in the scale of the reconstruction. So, in order to compare them, the relative scale s is estimated first by minimizing the error between the two trajectories. The Bundle Adjustment trajectory is then scaled $\mathbf{x}_{BA}^{C_1} = f_{scale}\left(\mathbf{x}_{BA}^{C_1} \right)$ and also its covariance $P_{BA}^{C_1} = J_{scale} P_{BA}^{C_1} J_{scale}^\top$.

Finally, the error is computed as the relative transformation between the two solutions:

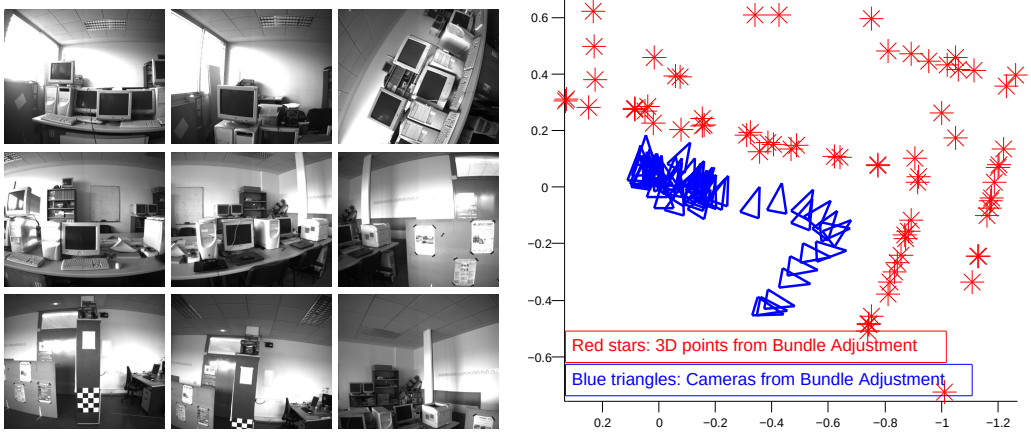
$$e = \oplus \mathbf{x}_{BA}^{C_1} \ominus \mathbf{x}_{MS}^{C_1} \; ; \tag{25}$$

and the corresponding covariance of the error is computed by propagating the covariances of the global optimization and sequential estimate:

$$P_e = J_{eBA} P_{BA}^{C_1} J_{eBA}^\top + J_{eMS} P_{MS}^{C_1} J_{eMS}^\top \; . \tag{26}$$

It was checked in the experiments in the paper that the covariance term from Bundle Adjustment, $J_{eBA} P_{BA}^{C_1} J_{eBA}^\top$, was negligible with respect to the summed covariance P_e. Since this is the case, it is our opinion that the Bundle Adjustment results can be considered as a reliable ground truth to evaluate sequential approaches. In the following figures, only uncertainty regions coming from filtering, $J_{eMS} P_{MS}^{C_1} J_{eMS}^\top$ are shown.

The same subsampled sequence was used for all the experiments in the following Sections 5.2 and 5.3. The camera moves freely in six degrees of freedom in a computer lab, with the maximum distances between camera locations around 5 metres. Filter tuning parameters were equal for all the experiments: motion dynamic and



(a) Sample images from the sequence used for benchmarking.

(b) Reference solution computed by the proposed method.

Figure 4: Images extracted from the sequence used in the experiments and reference camera positions extracted.

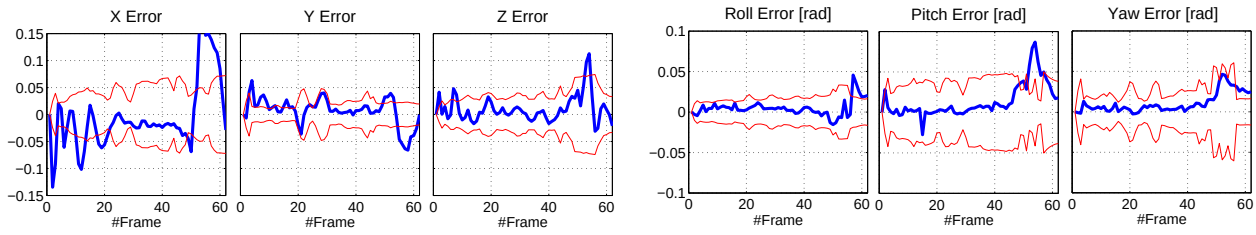
measurement model noise were kept the same, the number of measured features in the image was limited to 30 and all the thresholds (e.g. for feature deletion, cross-correlation, inverse depth to Euclidean conversion and initialization) were also kept the same. The reader should be aware that despite all of care taken, the experiments are not exactly the same: One of the reasons is that the outlier rate is different for each method; some methods need to initialize more features in order to keep measuring 30. Nevertheless, in the opinion of the authors, this is the fairest comparison as the algorithms try always to measure always the same number of points and hence gather an equivalent amount of sensor data.

Figure 4 shows example images from the sequence used in the following two sections for 1-point RANSAC and JCBB benchmarking. The 62 camera locations from the 2796 images long sequence are also displayed. Results for different experiments using this benchmarking method have been grouped for better visualization and comparison: Figures 5 and 7 show estimation errors for different tunings of 1-point RANSAC and JCBB; and 9 details their computational cost. All the experiments in the paper were run on an Intel(R) Core(TM) i7 processor at 2.67GHz.

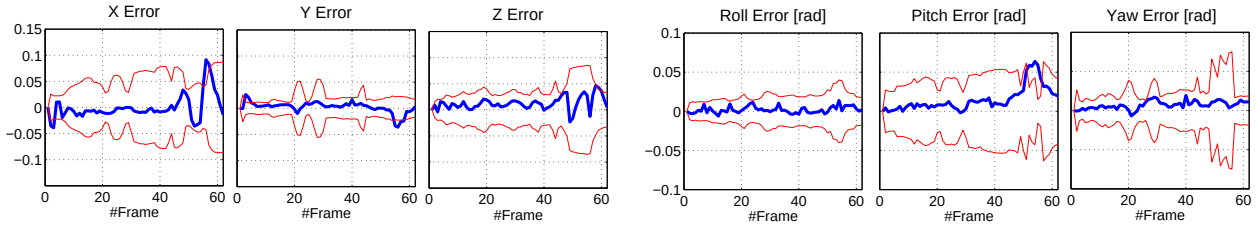
5.2 1-Point RANSAC

First, the performance of 5-point and 1-point RANSAC is compared, in order to ensure that there is no degradation of performance when the sample size is reduced. Figures 5(a) and 5(b) show the errors of both algorithms with respect to the reference camera motion, along with their 99% uncertainty regions. It can be observed that reducing the sample size from 5 to 1 does not have a significant effect either on the accuracy or the consistency of the estimation. On the contrary, the figure even shows 1-point outperforming 5-point RANSAC. We attribute this to the fact that the theoretical number of hypotheses given by equation 1 was not inflated in our experiments, unlike in classical SfM algorithms (Raguram et al., 2008). By increasing the number of iterations, 5-point RANSAC results comes close to 1-point; but we find it remarkable that without this augmentation 1-point RANSAC already shows good behaviour. The standard deviation of image noise was chosen to be 0.5 for the experiments, as subpixel matching is used.

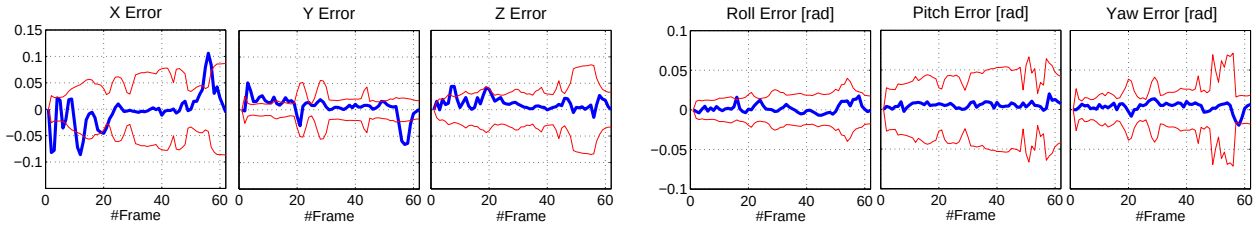
While the accuracy and consistency remains similar, the computational cost is much higher for the usual



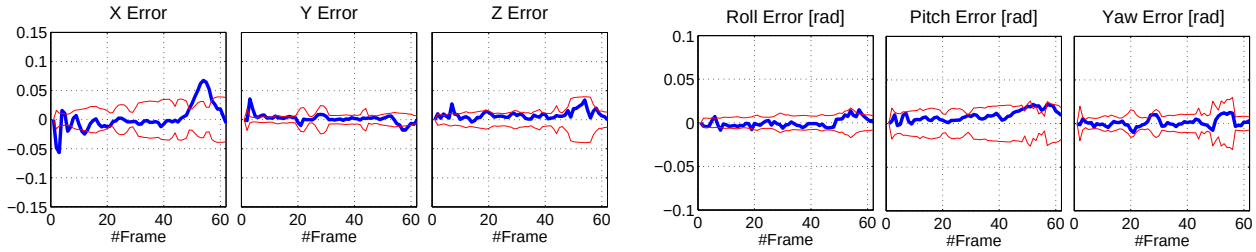
(a) 5-point RANSAC, $\sigma_z = 0.5$ pixels



(b) 1-point RANSAC, $\sigma_z = 0.5$ pixels

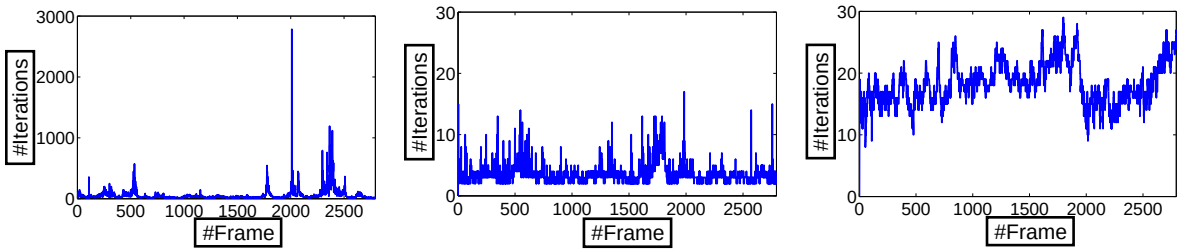


(c) 1-point exhaustive hypothesis, $\sigma_z = 0.5$ pixels



(d) 1-point RANSAC, $\sigma_z = 0.2$ pixels

Figure 5: Camera location error (in thick blue) and uncertainty (in thin red) for different RANSAC configurations. Similar error and consistency is shown for 5-point and 1-point RANSAC in Figures 5(a) and 5(b) respectively. Figure 5(c) also reports similar results for exhaustive hypothesis testing. Figure 5(d) shows smaller errors as a result of making 1-point RANSAC stricter by reducing the standard deviation of measurement noise.



(a) Number of iterations along the sequence for 5-point RANSAC.

(b) Number of iterations along the sequence for 1-point RANSAC.

(c) Number of iterations along the sequence for exhaustive hypotheses generation.

Figure 6: Number of iterations for 5-points and 1-point RANSAC. Notice the several orders of magnitude increse for the 5-point case, causing a large cost overhead when compared with 1-point RANSAC (Figures 9(a), 9(b) and 9(c) detail the computational cost for the three cases respectively).

5-point RANSAC than the proposed 1-point. The detail of the computational cost of both algorithms can be seen in Figures 9(a) and 9(b). The cost of RANSAC is low compared with the rest of the EKF computations for the 1-point case, but it is several orders of magnitude higher and is the main cost in the 5-point case. The large cost increase is caused by the increase in the number of random hypotheses in frames with a large number of spurious matches. Figures 6(a) and 6(b) show the number of hypotheses in both cases, revealing that in 5-point RANSAC this is two orders of magnitude. The five higher green pikes appearing in all the figures are caused by dropped frames in the sequence where there is a jump in camera location. The correspondence search cost is increased at these frames, but notice that the cost of RANSAC is not increased at all.

Hypothesis generation from a single point opens the possibility of exhaustive rather than random hypotheses generation: while an exhaustive generation of all the possible combinations of 5 points in the measurement subset would be impractical, an exhaustive generation of 1-point hypotheses implies only as many hypotheses as measurements. Figure 5(c) details the errors for the 1-point exhaustive hypotheses generation case. Compared with 1-point random hypotheses generation in Figure 6(b), we observe similar accuracy and consistency. Figure 6(c) shows the number of iterations needed for comparison with the random adaptive case (Figure 6(b)). The computational cost is increased but, as shown in Figure 9(c), it is still dominated by the EKF update cost. Both options are then suitable for real-time implementation, with the cheaper adaptive random 1-point RANSAC algorithm being preferable as performance is not degraded significantly.

From analyzing the computational cost in Figure 9(b) it can be concluded that the cost for 1-point RANSAC is always low compared with EKF computation even when the spurious match rate is high (the spurious match rate is shown in Figure 8(b)). As will be shown later, the latter becomes an important advantage over JCBB whose cost grows exponentially with the rate of spurious matches. This efficiency opens the possibility of making the RANSAC algorithm stricter by reducing the measurement noise standard deviation and hence discarding high noise points in the EKF. Such analysis can be done by reducing the standard deviation from 0.5 to 0.2 pixels: high noise points were discarded as outliers, as can be seen in Figures 8(b) and 8(d). The computational cost increases, as shown in Figure 9(e), but still remains small enough to reach real-time performance at 22 Hz. The benefit of discarding high noise points can be observed in Figure 5(d): errors and their uncertainty were reduced (but still kept highly consistent) as a result of measuring more accurate points.

5.3 JCBB

RANSAC and JCBB tuning is a thorny issue when benchmarking both algorithms. As both cases assume Gaussian distributions for the measurement and decide based on probability, we considered it fairest to choose equal significance levels for the probabilistic tests of both algorithms. The significance level was chosen to be 0.05 in the χ^2 test that JCBB performs to ensure joint compatibility for the matches. Consistently, the probabilistic threshold for RANSAC was set to 95% for voting (line 15 in the algorithm in Section 3) and for the rescue of high-innovation matches (line 29 in the algorithm in Section 3).

The results of benchmarking JCBB are shown in the following figures. First, Figure 7(a) details the errors and uncertainty regions for the EKF using JCBB. It can be observed that the estimation in Figure 7(a) show larger errors and inconsistency than the 1-point RANSAC one in Figure 7(b), repeated here for visualization purposes. The reason can be observed in Figure 8, where the outlier rates for 1-point RANSAC and JCBB are shown: the number of matches considered outliers by 1-point RANSAC is greater than by JCBB. The points accepted as inliers by JCBB are the ones that spoil the estimation.

A stricter version of JCBB has been benchmarked by reducing the standard deviation of uncorrelated measurement noise to 0.2 pixels, as was done with 1-point RANSAC. The spurious match rate for both algorithms, shown in Figure 8(c) and 8(d), shows that 1-point RANSAC remains more discriminative and hence produces more accurate estimation than JCBB (Figure 7(c)). 1-point RANSAC errors for the same tuning are repeated in 7(d) for comparison purposes. Also, as previously noted, the computational cost of JCBB grows exponentially when made stricter: Figure 9(f) shows peaks over a second in the worst cases.

JCBB can also be made stricter by increasing the significance level α of the χ^2 test it performs to check the joint compatibility of the data. Several experiments were run varying this parameter. The lowest estimation errors, shown in Figure 7(e), were reached for $\alpha = 0.5$ instead of the usual $\alpha = 0.05$. Estimation errors for this best JCBB tuning are still larger than in any of the 1-point RANSAC experiments.

5.4 Trajectory Benchmarking against GPS.

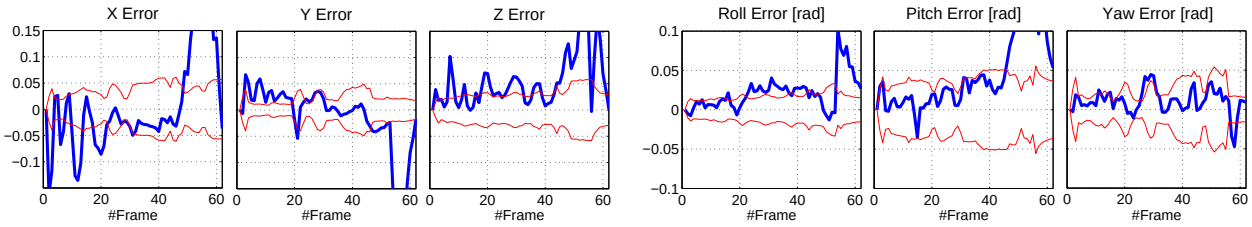
The following sections benchmark the presented filtering scheme for the estimation of long camera trajectories. The benchmarking method of the previous section becomes difficult to apply here, so camera translation only is benchmarked against GPS data. This section describes the benchmarking procedure.

Similarly to the previous section, our EKF estimation takes the first camera frame C_1 as the frame of reference. A similarity transformation (rotation $\mathbf{R}^W_{C_1}$, translation $\mathbf{t}^W_{C_1}$ and scale s) has to be applied which aligns every point of the trajectory $\mathbf{r}^{C_1}_{C_k} = \begin{bmatrix} x^{C_1}_{C_k} & y^{C_1}_{C_k} & z^{C_1}_{C_k} \end{bmatrix}^\top$ with the GPS data $\mathbf{r}^W_{GPS_k}$, whose frame of reference we will denote by W:

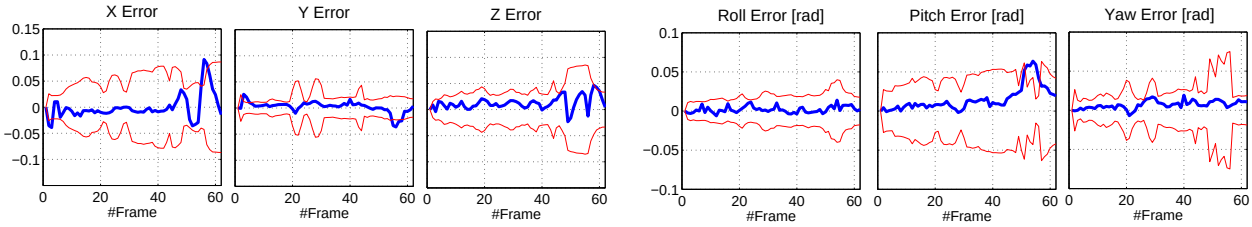
$$\begin{bmatrix} \mathbf{r}^W_{C_k} \\ 1 \end{bmatrix} = \begin{bmatrix} x^W_{C_k} \\ y^W_{C_k} \\ z^W_{C_k} \\ 1 \end{bmatrix} = \begin{bmatrix} s\mathbf{R}^W_{C_1} & \mathbf{t}^W_{C_1} \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} x^{C_1}_{C_k} \\ y^{C_1}_{C_k} \\ z^{C_1}_{C_k} \\ 1 \end{bmatrix} . \tag{27}$$

The value of $\mathbf{t}^W_{C_1}$ is taken from the GPS data in the first camera frame. Trajectory estimation from pure monocular vision will not be able to recover the scale s, which will remain unknown. For the combination of a monocular camera and wheel odometry input, the overall scale of the estimation is observed by odometry readings and then $s = 1$ in Equation 27. The rotation between GPS and the first camera position $\mathbf{R}^W_{C_1}$ will be unknown in every case, as it is non-observable from GPS readings.

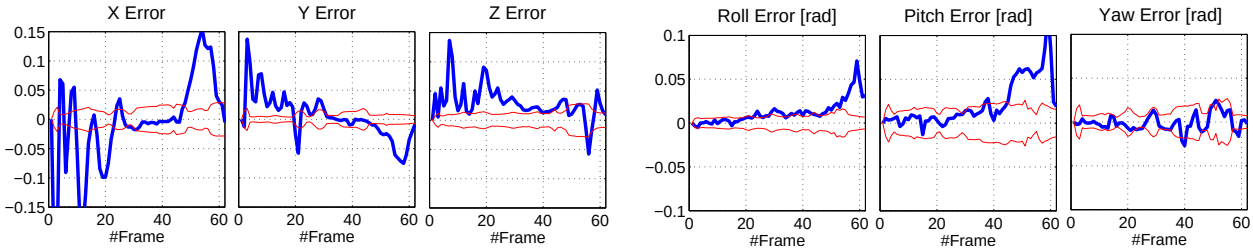
The unknown parameters of the alignment (s and $\mathbf{R}^W_{C_1}$ for pure monocular, and only $\mathbf{R}^W_{C_1}$ for monocular plus



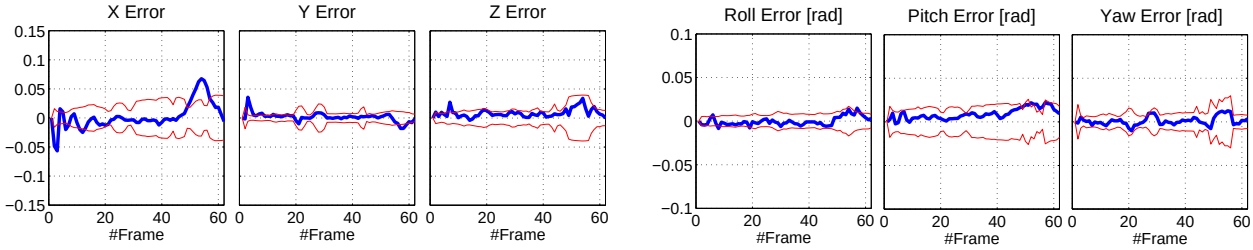
(a) JCBB, $\sigma_z = 0.5$ pixels



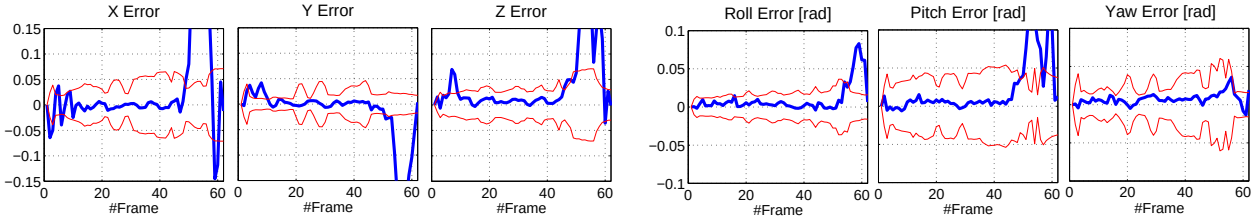
(b) 1-point RANSAC, $\sigma_z = 0.5$ pixels



(c) JCBB, $\sigma_z = 0.2$ pixels

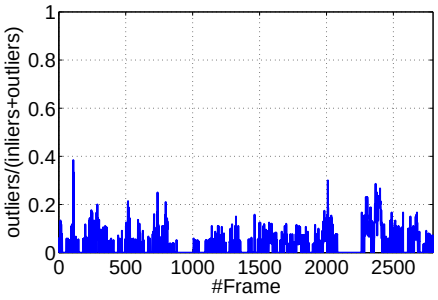


(d) 1-point RANSAC, $\sigma_z = 0.2$ pixels

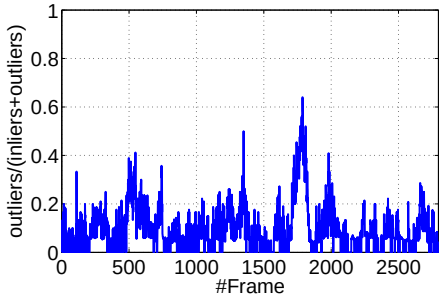


(e) JCBB, $\sigma_z = 0.2$ pixels, $\alpha = 0.5$

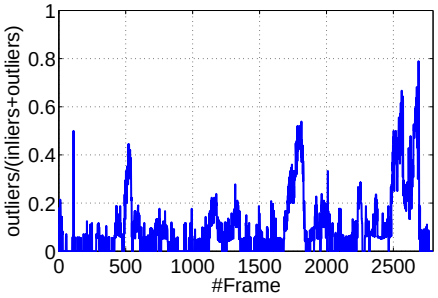
Figure 7: Camera location errors when using JCBB is shown in Figures 7(a) and 7(c), for standard deviations of 0.5 and 0.2 pixels respectively. Figures 7(b) and 7(d) showing 1-point RANSAC results for the same filter tuning are repeated here for comparison. It can be seen that 1-point RANSAC outperforms JCBB in both cases. Figure 7(e) shows the best JCBB tuning found by the authors, which still gives worse results than 1-point RANSAC.



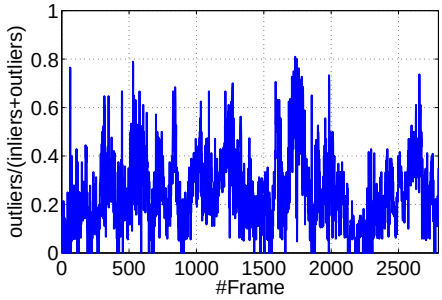
(a) JCBB, $\sigma_z = 0.5$ pixels.



(b) 1-point RANSAC, $\sigma_z = 0.5$ pixels

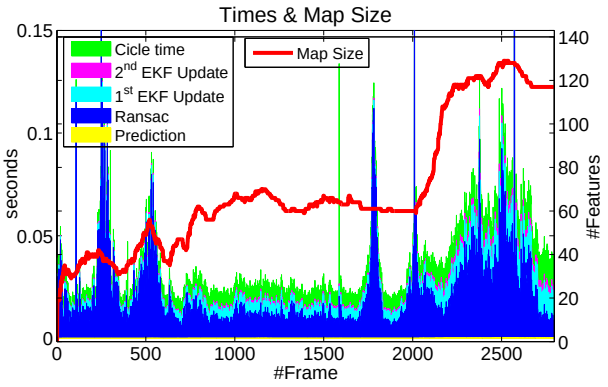


(c) JCBB, $\sigma_z = 0.2$ pixels

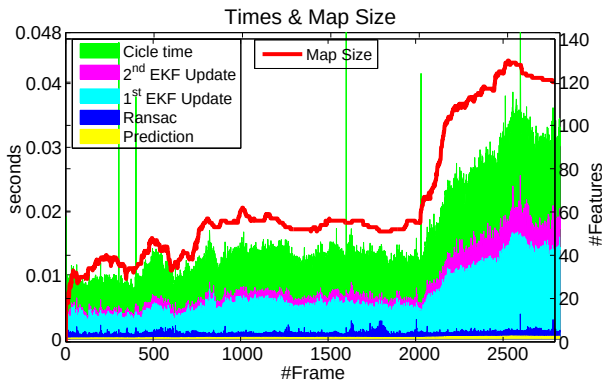


(d) 1-point RANSAC, $\sigma_z = 0.2$ pixels.

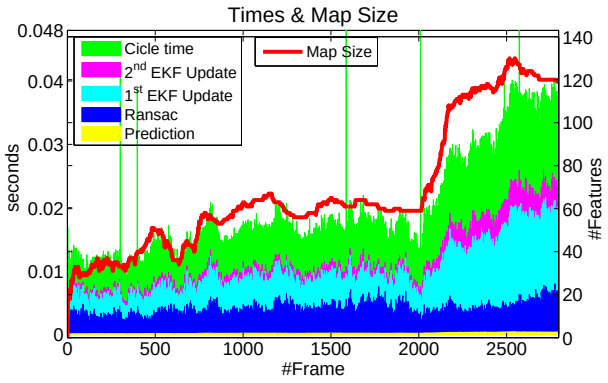
Figure 8: Spurious match rate for JCBB and RANSAC when measurement noise standard deviation σ_z is reduced to 0.2 pixels. It can be observed that reducing the measurement noise makes both techniques stricter, but 1-point RANSAC remains more discriminative.



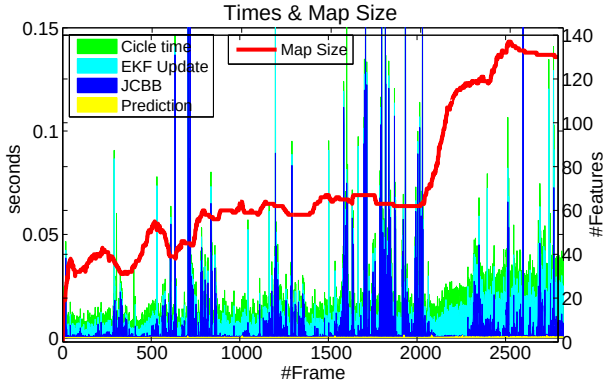
(a) 5-point RANSAC, $\sigma_z = 0.5$ pixels



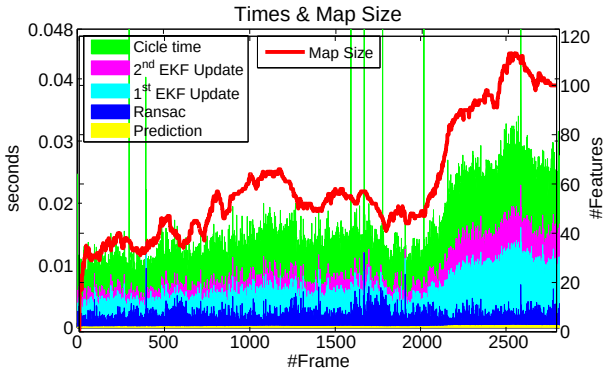
(b) 1-point RANSAC, $\sigma_z = 0.5$ pixels



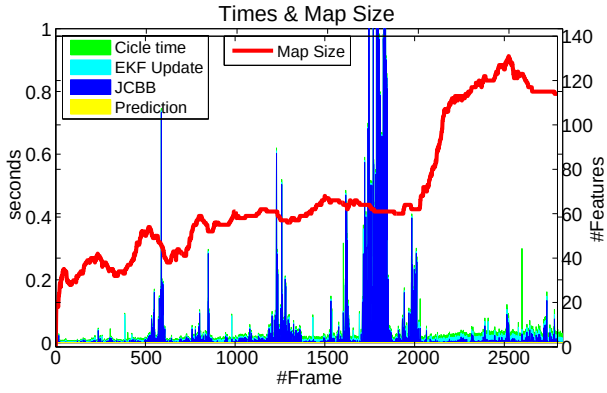
(c) 1-point exhaustive hypothesis, $\sigma_z = 0.5$ pixels



(d) JCBB, $\sigma_z = 0.5$ pixels



(e) 1-point RANSAC, $\sigma_z = 0.2$ pixels



(f) JCBB, $\sigma_z = 0.2$ pixels

Figure 9: Detail of times and map sizes for different RANSAC and JCBB configurations in double y-axis figures: times are shown as areas and measured in seconds on the left y-axis; the map size is displayed as a a red line and is measured on the right y-axis. 1-point RANSAC exhibits much lower computational cost than 5-point RANSAC and JCBB. 1-point RANSAC also shows only a small increase when made exhaustive or stricter, making it suitable for real-time implementation at 22 Hz for the map size detailed in the figures.

wheel odometry) are obtained via a non-linear optimization that minimizes the error between the aligned trajectory $\mathbf{r}^W_{C_k}$ and the GPS trajectory $\mathbf{r}^W_{GPS_k}$.

For the sake of simplicity, the assumption that the position of the camera sensor and the GPS antenna coincide on the robot has been made in the above reasoning, which is reasonable as the position of the sensors differ by only a few centimetres and robot paths cover hundreds of 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:

$$e_k = \sqrt{\left(\mathbf{r}^W_{C_k} - \mathbf{r}^W_{GPS_k}\right)^\top \left(\mathbf{r}^W_{C_k} - \mathbf{r}^W_{GPS_k}\right)}. \tag{28}$$

5.5 Pure Monocular EKF-Based Estimation for Long Sequences

Three different sequences from the *RAWSEEDS* dataset have been used to test the validity of the 1-point RANSAC EKF for long-term camera motion estimation. All sequences were recorded by a 320×240 Unibrain camera with a wide-angle lens capturing at 30 fps.

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 long and 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 trajectory, the relative error with respect to the trajectory keeps the same low value as the other two shorter sequences (1% of the trajectory length).

Figure 10 shows an image from the 650 metres experiment, along with the tracked features. It can be observed that around a hundred features per frame had to be measured in order to reduce scale drift error. This high number will increase the computational cost of the EKF beyond real-time bounds for the pure monocular case. In the particular experiments presented, the algorithm runs at about 1 Hz. Nevertheless, it will be shown in next subsection how introducing extra information about the scale will reduce the number of measurements, enabling real-time performance for the combination of visual tracking plus wheel odometry.

Figure 11 shows the estimated (in black) and the GPS (in red) trajectories over a top view extracted from Google Maps for each one of the sequences. The accuracy of the estimated trajectories is clear from visual inspection. Table 1 details the maximum and mean errors obtained in these experiments and also for the experiment in the next section combining monocular vision and wheel odometry inputs. Figure 12 shows histograms of the errors for the three sequences.

Subfigures 12(c) and 12(d) in this latter figure show histograms of the errors for the 650 metres experiment in two different versions of the 1-point RANSAC algorithm: the first one of them using the algorithm 1 and the second one replacing the random hypotheses generation with exhaustive hypotheses generation as evaluated in Figure 5(c). The conclusion from section 5.2 is confirmed here: exhaustive hypothesis generation only very slightly improves the estimation errors; so adaptive random 1-point RANSAC should be preferred.

5.6 Visual Odometry from a Monocular Sequence plus Wheel Odometry

Figure 13 shows the trajectory obtained by the visual odometry algorithm over a GoogleMaps plot and compared against GPS data. The length of the estimated trajectory is about 1310 metres and was covered by the RAWSEEDS mobile robot in 30 minutes, capturing 54000 frames. The maximum and mean error were 23.6 and 9.8 metres respectively. Adding wheel odometry information allowed us to reduce the number of tracked features to 25, enabling real-time operation at 30 frames per second.

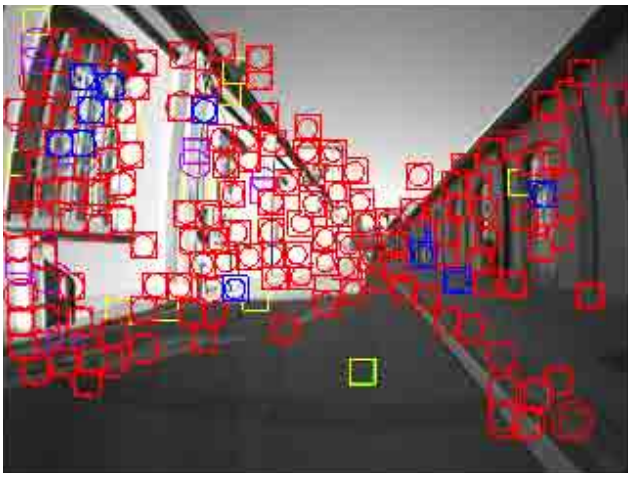
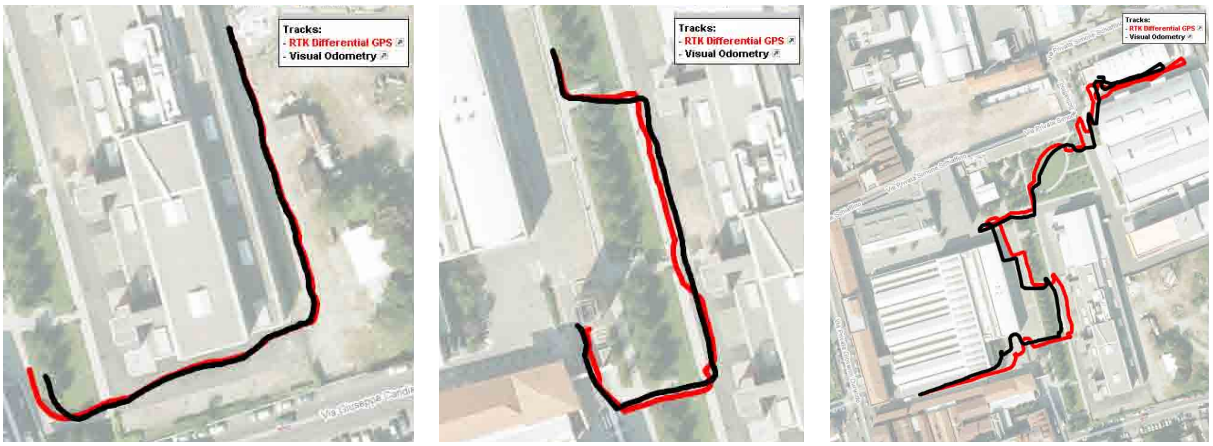


Figure 10: Image from the 650 metres sequence, showing the high number of tracked features.

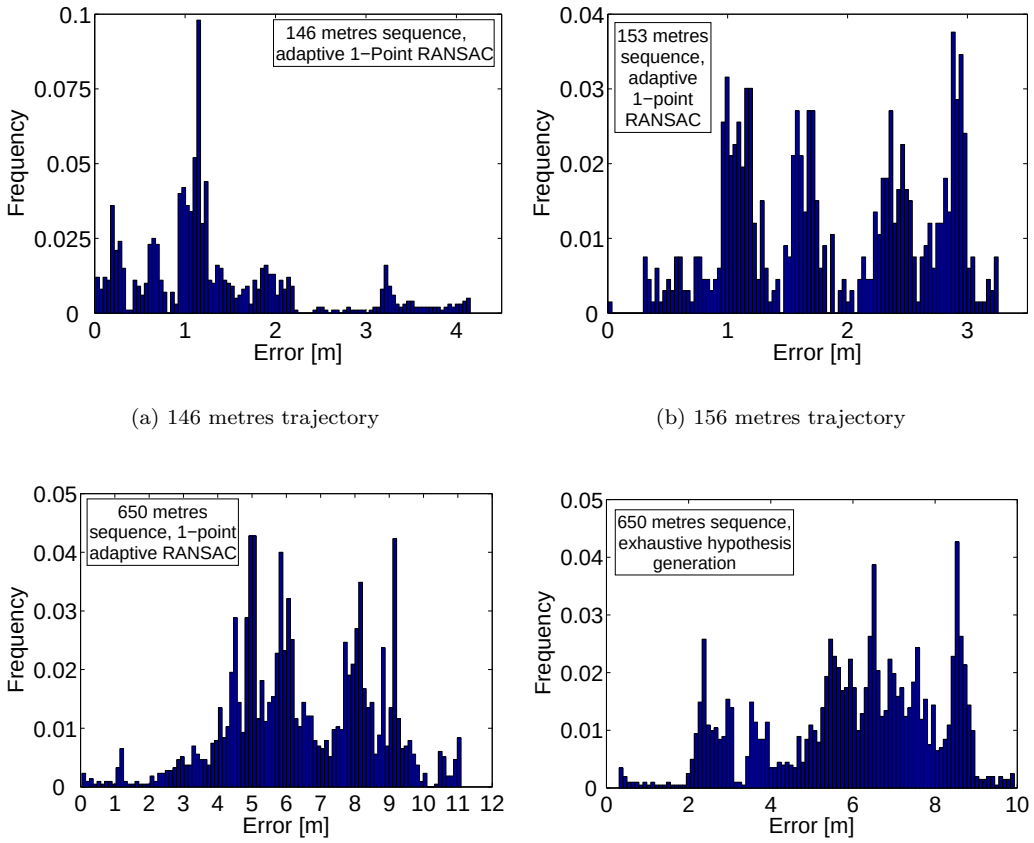


(a) 146 metres trajectory (b) 156 metres trajectory (c) 650 metres trajectory

Figure 11: Estimated trajectories from pure monocular data and GPS data

Table 1: EKF-based visual estimation error for long camera trajectories.

Trajectory length [m]	Sensor used	Mean error [m]	Maximum error [m]	% mean error over the trajectory
146	monocular	1.3	4.2	0.9%
153	monocular	1.9	3.3	1.1%
650	monocular	6.4	11.1	1.0%
1310	monocular and wheel odometry	9.8	23.6	0.7%



(a) 146 metres trajectory

(b) 156 metres trajectory

(c) 650 metres trajectory

(d) 650 metres trajectory; Exhaustive-SAC

Figure 12: Histograms of the errors for the three experiments using only monocular information

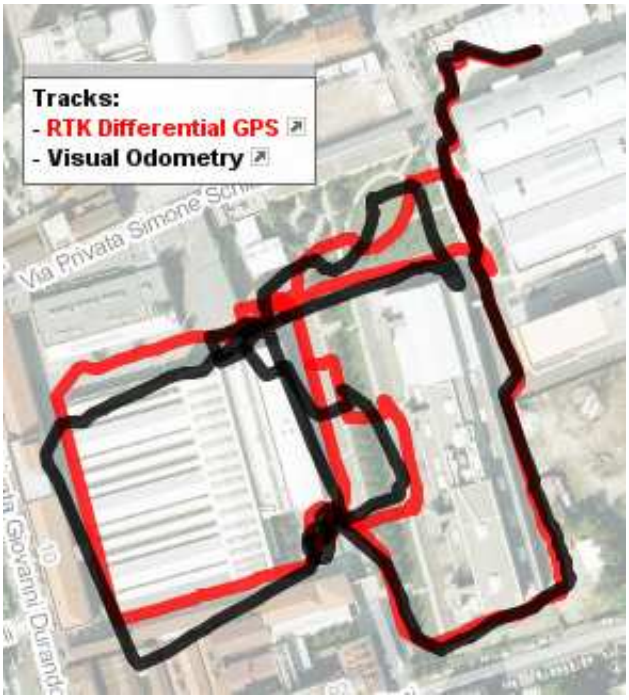


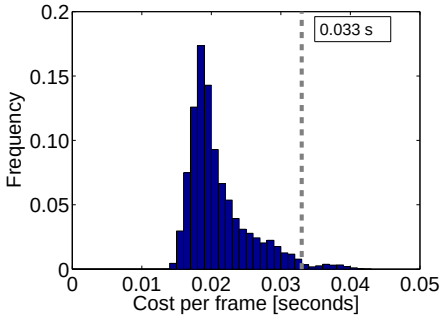
Figure 13: Visual odometry results compared against RTK GPS over a Google Maps plot.

The processing time per frame for this sequence using 1-point RANSAC can be observed in Figure 14 in the form of a histogram. It can be noticed that the total computational cost per step is under 33 milliseconds in 98% of the frames, suggesting that the algorithm is suitable for real-time implementation. It can be observed in the right-hand figure that for the same number of image measurements JCBB's computational cost far exceeds real-time constraints in a large number of frames. JCBB's exponential complexity arises in this experiment in frames where a significant proportion of outliers are present, expanding the tail of the histograms of the figure. For this particular experiment, JCBB's histogram expands to 2.4 seconds while 1-Point RANSAC's maximum time only reaches 0.44 seconds.

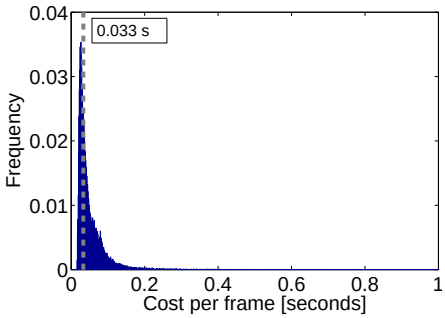
Figure 14 also shows two histograms representing the computational cost of both algorithms when the number of features in the image is increased to 50. It can be observed that the cost of 1-Point RANSAC grows, but still the processing cost is always on the order of tenths of a second. JCBB's cost reaches maximum values of several hours, and processing times of several seconds per frame are not unusual

Figure 15(b) shows raw odometry as a red thin line and GPS with a blue thick line for comparison. It can be observed that early drift appears and the plotted trajectory is rather far from the GPS locations. Figure 15(a) shows pure monocular estimation in thin red and GPS measurements in thick green. Observing this plot carefully, it can be observed that a monocular camera is able to very accurately estimate orientation, but the unobservability of the scale produces drift in this parameter for the number of tracked features (25) considered in this experiment.

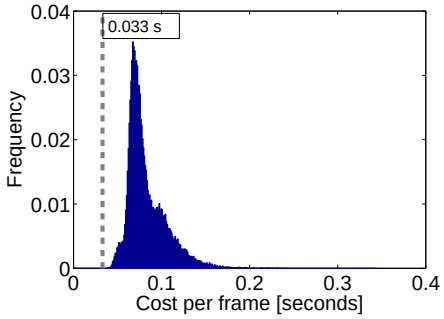
Finally, Figure 15(c) details the estimated trajectory that can be achieved from the combination of the two sensors. Accurate estimation is achieved for a trajectory of 1.3 kilometres, which can be compared with state of the art in monocular visual odometry, e. g. (Scaramuzza et al., 2009).



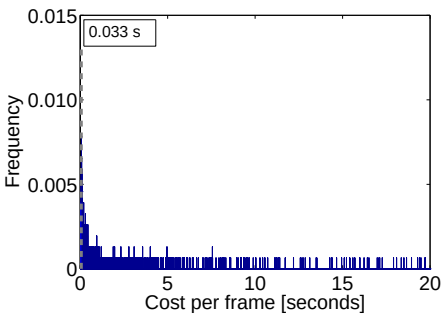
(a) 1-point RANSAC; 25 measured features per frame



(b) JCBB; 25 measured features per frame

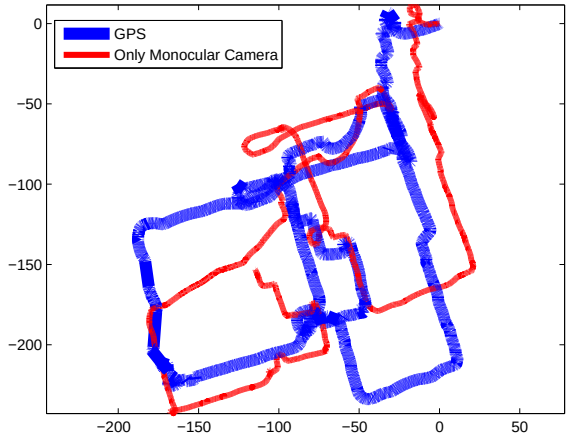


(c) 1-point RANSAC; 50 measured features per frame

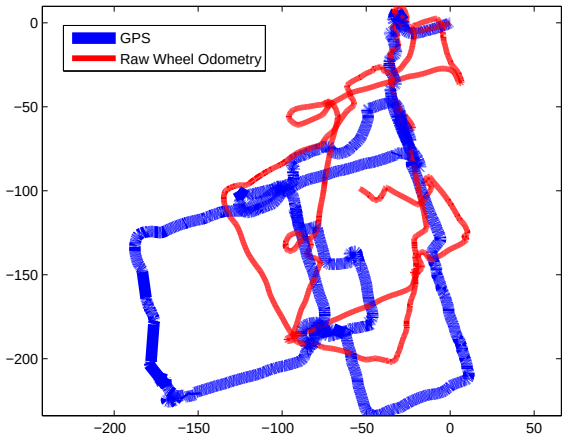


(d) JCBB; 50 measured features per frame

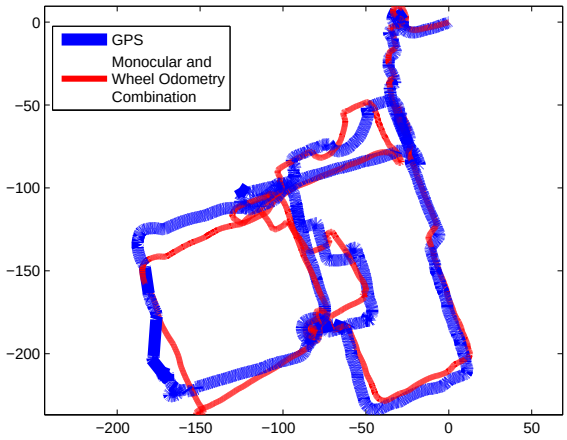
Figure 14: Histograms showing the computational cost for RANSAC and JCBB for the cases of 25 and 50 image points per frame. Experiment 14(d) had to be early terminated at frame 1533, as JCBB computational cost rises in some frames up to 1544 seconds



(a) Pure monocular estimation (thin red) tracking 25 features and GPS trajectory (thick blue). Large errors appear caused by scale drift, which is unobservable by a monocular camera.



(b) Raw odometry measurements (thin red) and GPS trajectory (thick blue). Errors in raw odometry are caused by early drift typical from proprioceptive sensors.



(c) Visual Odometry estimation from the combination of monocular camera plus wheel odometry (thin red) and GPS trajectory (thick blue). The combination of both sensors overcomes their deficiencies when used alone. Real-time performance at 30 Hz can be achieved, and error is 0.7% of the trajectory.

Figure 15: Pure monocular estimation showing scale drift in Figure 15(a), raw odometry input showing drift in Figure 15(b) and visual odometry results combining the two in Figure 15(c); all are compared against GPS trajectory (thick blue line).

6 Discussion

While the relevance of algorithms like JCBB or the recent Active Matching (AM) reside on their generality, the main advantage in the presented approach is its efficiency. 1-point RANSAC is directed to the particular case of a rigid scene. The rich variety of correlation patterns that a covariance matrix can encode is manageable by general methods like JCBB or AM. Our 1-point RANSAC exploits the very simple pattern where all the correlations are mainly explained by sensor motion, and hence small size data subsets are enough to constraint the rest of the measurements. For more complex models, like non-rigid scenes or multi-object tracking, 1-point RANSAC may not offer such a satisfactory result.

Nevertheless, it is also true that estimation from a moving sensor's data stream in an almost rigid scene covers a great percentage of SLAM problems; and a specific method more efficient than general methods can be of importance. In this sense, 1-point RANSAC outperforms existing approaches by presenting lower cost and scaling well with the state vector and measurement size, and also with the outlier rate. The computational overhead it introduces is always smaller than 10% of standard EKF's computational cost, such that it can be readily used in any existing algorithm. Visual EKF SfM, already proven to run in real-time, still keep real-time performance and provides the benefit in accuracy of spurious match rejection when 1-point RANSAC is used.

Besides its efficiency, 1-point RANSAC also has some advantages in dealing with non-linearities as a result of checking rigidity after data fusion where some of the inaccuracies introduced by non-linearities have been compensated. This advantage is shared with Active Matching. On the contrary JCBB checks rigidity before data fusion, which is a serious drawback of the algorithm.

Since 1-point RANSAC is able to deal with large outlier rates at low computational overhead, we find it interesting to force the EKF into a low measurement error operation mode. For a small cost increase, the EKF is fed only very accurate measurements (selected by "a survival of the fittest" process, where the fittest features are those producing the lowest error measurements) and hence the accuracy of the estimation is improved as seen in Figure 5(d). This particular operation mode can only be achieved due to the efficiency of the presented algorithm, being impractical if spurious match rejection is expensive.

It is also worth remarking that although this paper is focused on the particular case of EKF visual estimation, the new 1-point RANSAC method presented here is independent of the type of sensor used. The only requirement is the availability of highly correlated prior information, which is typical of EKF SLAM for any kind of sensor used — and also in the multisensor case. Also, as highly correlated priors are not exclusive to EKF SLAM, the applicability of 1-point RANSAC could be even broader. As an example, we think that camera pose tracking in keyframe schemes (Klein and Murray, 2008; Mouragnon et al., 2009) would benefit from our 1-point RANSAC cost reduction if a dynamic model were added to predict camera motion between frames.

7 Conclusions

A novel RANSAC algorithm is presented in this paper which, for the first time and differently from standard purely data-driven RANSAC, incorporates *a priori* probabilistic information into the hypothesis generation stage. As a consequence of using this prior information, the sample size for the hypothesis generation loop can be reduced to the minimum size of 1 point data. 1-point RANSAC has two main strengths worth summing up here. First, as in standard RANSAC, model constraints are checked *after* hypothesis data has been fused with the a priori model, an advantage over JCBB. Second, using 1-point plus prior knowledge hypotheses greatly reduces the number of hypotheses to construct and hence the computational cost compared with usual RANSAC based solely on data. Its linear cost in the state size also outperforms JCBB's exponential complexity in the number of outliers. In a practical sense, its linear complexity means an overhead of less than 10% of the standard EKF cost, making it suitable for real-time implementation in local visual SLAM

or SfM.

The paper presents a method for benchmarking six degrees of freedom camera motion estimation results. The method presents three clear advantages: First, it is intended for real image sequences and includes effects difficult to reproduce by simulation (like non-Gaussian image noise, shaking handy motion, image blur or complex scenes). Second, it is easily reproducible as the only hardware required is a high resolution camera. And third, the effort required by the user is low. The uncertainty of the estimated solution also comes as an output of the method and the appropriateness of Bundle Adjustment estimation as reference can be validated. The method has been used to prove the claimed superiority of the 1-point RANSAC method described in the paper.

The general EKF plus 1-point RANSAC algorithm has been also experimentally tested for the case of large camera trajectories in outdoor scenarios. Sensor-centered filtering instead of the traditional world-centered method has been used in order to reduce the uncertainty in the area local to the current camera and reduce linearization errors. For the pure monocular case, errors around 1% of the trajectory have been obtained for trajectories up to 650 metres from a publicly available dataset. The number of tracked features in the image has to be increased to $100 - -200$ in order to avoid scale drift. This high number makes this case currently moves us away from real-time performance, and the method runs at 1 frame per second.

The combination of monocular vision and wheel odometry has also been benchmarked for the visual odometry application. The extra odometric information makes scale observable; the number of tracked features can be reduced and real-time performance can be achieved for this case. A 1300 metre long trajectory has been estimated in the paper, with the mean error against GPS coming out at 0.7% of the trajectory.

8 Future Work

Having already evaluated 1-Point RANSAC's performance against the gold-standard JCBB, it would be very interesting to compare it against the most recent algorithms for spurious rejection when using Extended Kalman Filter. Particularly, Active Matching (Chli and Davison, 2008) and Randomized Joint Compatibility (RJC) (Paz et al., 2008), described with detail in the related work section, are seen by the authors as the most relevant works on the topic. It is the authorss opinion that randomized versions of both algorithms could come close to 1-point RANSAC at the cost of losing their generality.

The basic idea of using 1-point hypotheses has been presented in this paper in its most basic form. But it does not have any incompatibility with the recent techniques described in the related work section that aim to rapidly discern good and bad hypotheses and lower the computational cost. Integrating 1-Point RANSAC with one or several of these ideas could lead to an even faster algorithm.

A very interesting thought for future work is to reduce even more the sample size; going from the presented 1-Point RANSAC to Half-A-Point RANSAC. Following the argument of this paper, the integration of a probabilistic prior and only one dimension of a 2D-measurement (in the visual estimation case) could be enough to provide a valid hypothesis for the model. This further sample size reduction would produce even higher computational savings.

Acknowledgments

This work was supported by projects DPI2009-07130 (Dirección General de Investigación of Spain), RoboEarth FP7-248942 (European Union) and European Research Council Starting Grant 210346. We are grateful to Belén Masiá, Brian Williams, Ian Reid, Frank Dellaert, J. Neira and J. D. Tardós for fruitful discussions; and to the University of Oxford (Ian Reid) and Imperial College for software collaboration. We also thank the anonymous reviewers for their thorough reviews which helped to improve the paper.

References

Blanco, J.-L., Moreno, F.-A., and González, J. (2009). A collection of outdoor robotic datasets with centimeter-accuracy ground truth. *Autonomous Robots*, 27(4):327–351.

Borenstein, J., Everett, H., and Feng, L. (1996). Where am I? Sensors and methods for mobile robot positioning. *University of Michigan*, 119:120.

Capel, D. (2005). An effective bail-out test for ransac consensus scoring. In *Proceedings of the British Machine Vision Conference*, pages 629–638.

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

Cheng, Y., Maimone, M., and Matthies, L. (2006). Visual odometry on the Mars exploration rovers-a tool to ensure accurate driving and science imaging. *IEEE Robotics and Automation Magazine*, 13(2):54–62.

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

Chum, O. and Matas, J. (2008). Optimal randomized RANSAC. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(8):1472–1482.

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

Civera, J., Grasa, O. G., Davison, A. J., and Montiel, J. M. M. (2009). 1-point RANSAC for EKF-based structure from motion. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3498–3504.

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

Comport, A., Malis, E., and Rives, P. (2007). Accurate quadrifocal tracking for robust 3d visual odometry. *Proceedings of the 2007 IEEE International Conference on Robotics and Automation*, pages 40–45.

Davison, A. J. (2003). Real-time simultaneous localisation and mapping with a single camera. In *Ninth IEEE International Conference on Computer Vision, 2003. Proceedings*, pages 1403–1410.

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

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

Fenwick, J., Newman, P., and Leonard, J. (2002). Cooperative concurrent mapping and localization. In *Proceedings of the 2002 IEEE International Conference on Robotics and Automation*, volume 2, pages 1810–1817.

Fischler, M. A. and Bolles, R. C. (1981). Random sample consensus, a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381 – 395.

Funke, J. and Pietzsch, T. (2009). A framework for evaluating visual slam. In *Proceedings of the British Machine Vision Conference*.

Handa, A., Chli, M., Strasdat, H., and Davison, A. J. (2010). Scalable active matching. to appear in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

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

Klein, G. and Murray, D. (2008). Improving the Agility of Keyframe-Based SLAM. In *Proceedings of the 10th European Conference on Computer Vision: Part II*, pages 802–815. Springer.

Konolige, K., Agrawal, M., and Sol, J. (2007). Large-scale visual odometry for rough terrain. In *International Symposium on Research in Robotics*.

Kummerle, R., Steder, B., Dornhege, C., Ruhnke, M., Grisetti, G., Stachniss, C., and Kleiner, A. (2009). On measuring the accuracy of SLAM algorithms. *Autonomous Robots*, 27(4):387–407.

Moreno-Noguer, F., Lepetit, V., and Fua, P. (2008). Pose Priors for Simultaneously Solving Alignment and Correspondence. In *Proceedings of the 10th European Conference on Computer Vision: Part II*, pages 405–418. Springer-Verlag Berlin, Heidelberg.

Mouragnon, E., Lhuillier, M., Dhome, M., Dekeyser, F., and Sayd, P. (2009). Generic and real-time structure from motion using local bundle adjustment. *Image and Vision Computing*, 27(8):1178–1193.

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

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

Nistér, D. (2005). Preemptive RANSAC for live structure and motion estimation. *Machine Vision and Applications*, 16(5):321–329.

Nistér, D., Naroditsky, O., and Bergen, J. (2004). Visual odometry. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, volume 1, pages 652–659.

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

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

Raguram, R., Frahm, J., and Pollefeys, M. (2008). A Comparative Analysis of RANSAC Techniques Leading to Adaptive Real-Time Random Sample Consensus. In *Proceedings of the European Conference on Computer Vision*, pages 500–513.

RAWSEEDS (2010). *RAWSEEDS* public datasets web page. URL http://www.rawseeds.org/.

Scaramuzza, D., Fraundorfer, F., and Siegwart, R. (2009). Real-Time Monocular Visual Odometry for On-Road Vehicles with 1-Point RANSAC. In *Proceedings of the 2009 IEEE International Conference on Robotics and Automation*, pages 4293–4299.

Scharstein, D. and Szeliski, R. (2002). A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision*, 47(1):7–42.

Smith, M., Baldwin, I., Churchill, W., Paul, R., and Newman, P. (2009). The new college vision and laser data set. *The International Journal of Robotics Research*, 28(5):595 – 599.

Tardif, J., Pavlidis, Y., and Daniilidis, K. (2008). Monocular visual odometry in urban environments using an omnidirectional camera. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2531–2538.

Torr, P. and Murray, D. (1993). Outlier detection and motion segmentation. *Sensor Fusion VI*, 2059:432–443.

Torr, P. and Zisserman, A. (2000). MLESAC: A new robust estimator with application to estimating image geometry. *Computer Vision and Image Understanding*, 78(1):138–156.

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

Vedaldi, A., Jin, H., Favaro, P., and Soatto, S. (2005). KALMANSAC: Robust filtering by consensus. In *Proceedings of the International Conference on Computer Vision (ICCV)*, volume 1, pages 633–640.

Williams, B. (2009). *Simultaneous Localisation and Mapping Using a Single Camera*. PhD thesis, University of Oxford.

Williams, B., Klein, G., and Reid, I. (2007). Real-time SLAM relocalisation. In *IEEE 11th International Conference on Computer Vision*, page 1:8.