The 2010 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 18-22, 2010, Taipei, Taiwan

A New Approach to Vision-Aided Inertial Navigation

Jean-Philippe Tardif Michael George Michel Laverne
Alonzo Kelly Anthony Stentz
Robotics Institute, Carnegie Mellon University
10, 40th Street, Pittsburgh, PA, 15201, USA.
{tardifj,mgeorge,mlaverne,alonzo,axs}@rec.ri.cmu.edu

Abstract— **We combine a visual odometry system with an aided inertial navigation filter to produce a precise and robust navigation system that does not rely on external infrastructure. Incremental structure from motion with sparse bundle adjustment using a stereo camera provides real-time highly accurate pose estimates of the sensor which are combined with six degree-of-freedom inertial measurements in an Extended Kalman Filter. The filter is structured to neatly handle the incremental and local nature of the visual odometry measurements and to handle uncertainties in the system in a principled manner. We present accurate results from data acquired in rural and urban scenes on a tractor and a passenger car travelling distances of several kilometers.**

I. Introduction

A motion estimation device is one of the basic requirements for building an autonomous legged, wheeled or aerial robot. Besides autonomous navigation, other applications such as mapping and planetary landing also require accurate motion estimation [24], [29]. Such devices generally combine several sensors, which can be divided into two categories: *exteroceptive* and *proprioceptive*. Combining both types of sensors is an attractive solution because, roughly speaking, they have opposed strengths and weaknesses. The former estimate motion based on external observations such as images [8] or range data [20]. As a result, error accumulation or drift is essentially proportional to the length of the trajectory, although, it is also dependent on the geometry of the environment. The latter, by their nature, measure their own motion, and can operate in any kind of environment. The drawback is that error accumulation is a function of time rather than distance which is why they require some form of aiding.

With the recent advances in the manufacturing of micro-electro mechanical based inertial sensors (MEMS) and CCD sensors, it is possible to build inexpensive and reliable inertial measurement units (IMU) and cameras. As a result, vision-aided inertial navigation systems are increasingly popular. State of the art systems currently augment inertial measurements with visual odometry (VO) [1], [19], [21], [23].

II. Contributions

In this paper, we present a novel vision-aided navigation system which is based on an IMU and a stereo camera. Motion estimation is performed in real time at up to 30 frames per second and the integration of the sensors is robust. By *robust*, we imply that VO is allowed to fail and be restarted at any moment. This sometimes happens in real world situations, for example, if the camera is directly pointed at the sun, or when illumination quickly changes. In agricultural environments (see our results section), this can also occur when branches of a tree are too close to the camera. Note that the VO is also robust in that it can discard matching errors. It is also robust to other outliers due to non-stationary objects as demonstrated by our results in urban environements.

We rely on a delayed state Extended Kalman Filter (EKF) allowing a loose coupling of the two sensors. The delayed states simply correspond to the position and orientation of the vehicle pose at the last VO key frame. This effectively allows the VO input of the EKF to be a relative pose update. This has important practical advantages:

- The camera trajectory does not need to be registered within the global coordinate system;
- In case of failure, the update is set to have infinite uncertainty and the VO is simply restarted;
- Expensive uncertainty propagation over time [8], [11] is avoided since the uncertainty estimation is only required for the motion update.

Our visual-odometry is based on incremental structure from motion with key frame selection and sparse local bundle adjustment [10], [34]. This approach can also be referred to as an *optimization approach to VO* (see discussion by Strasdat *et al.* on the advantages of this approach over filtering [30]). It can be used on any kind of robot or vehicle since it does not rely on a specific motion model, *i.e.* it estimates a six degree of freedom pose of the camera and does not use any kind of smoothing. In addition, it does not make any assumption about the geometry of the scene such as a flat ground plane [14]. Distinctions from previous approaches are discussed further in Section III.

Our main application is navigation in agricultural environments such as orange groves and we demonstrate accurate results in that context. In addition we also test the approach

This material is based upon research supported by USDA/CSREES under Award Number 2008-51180-04890.

on challenging sequences of up to 20 kilometers acquired in urban environments. We show very good results despite using a stereo camera with a, less than ideal, baseline of 12 cm (Compared to 50 cm in [19] and 24 cm in [17]). We compare our results with ground truth data from post-processed differential GPS of centimeter-level accuracy.

The following provides an overview of the paper. Related work is discussed in Section III. The VO and EKF are described in Sections IV and V respectively. Finally, experiments are described in Section VI followed by concluding remarks in Section VII.

III. Related work

Since there is a large literature on VO and inertial navigation, most of this section is devoted to related work on vision-aided inertial navigation. Generally speaking, an approach can be classified either as tightly or loosely coupled.

In a tightly coupled system, information from the EKF is used by the VO. For example, in one of the first systems relying on both vision and inertial, Bosse *et al.* [5] disambiguate the pose estimation based on the prediction of the filter. In others, the sensor states as well as the visual landmarks are jointly estimated [27], [28], [31], [6], [35]. This allows correlation between measurements to be taken into account. However, this is computationally expensive, so real-time performance is achieved at the expense of accuracy of the VO by reducing the number of visual landmarks. Another potential issue with tightly coupled solutions is the difficulty of handling large position jumps (say after GPS corrections) since they may destabilize the vision system. Closely related to our work, is the one of Mourikis and Roumeliotis where delayed states corresponding to previous poses are maintained [23]. Instead of estimating the location of the visual landmarks in the EKF, they do so by triangulation, and still properly handle correlation between landmarks and states. Similarly, our approach can also express constraints between multiple camera poses whilst keeping the VO completely decoupled from the EKF.

Loose coupling implies an EKF at the top level receiving only inputs from VO. This has the main disadvantage of ignoring correlation between internal states of the devices. A common approach is to modify the EKF to handle relative pose measurements or constraints from the VO. Roumeliotis proposes to compute displacement on images [29] and Diel *et al.* rely on epipolar geometry [9]. These constraints are computed on pairs of consecutive images to make sure the VO measurements are not correlated. Another solution is to use the vision sensor as a global positioning sensor which is possible when a known target lies in the field of view of the camera [7].

Konolige *et al.* argue that ignoring correlation is a price worth paying for a more accurate optimisation approach to VO [19]. Their approach is similar to ours in philosophy, but the design of their EKF is quite different. They rely on a cascaded EKF where a low-level EKF is used to process inertial measurements and a high-level EKF fuses VO with filtered inertial measurements. This allows them to perform predictions using the VO and corrections with the IMU. We adopt the opposite approach and predict with the IMU measurements in our EKF, since the IMU has the highest update rate. Furthermore, their approach assumes that the acceleration profile is of zero mean locally in order to estimate roll and pitch angles. As a consequence, the uncertainty needs to be artificially high to compensate for this assumption. In our formulation, the direction of gravity implicitly damps roll and pitch errors without any arbitrary assumptions. As a result, the computed uncertainties are closer to the truth.

IV. Visual Odometry

Our VO relies on a calibrated stereo camera. Traditionally, stereo cameras have been treated as range sensors. Thus, motion estimation involved dense or sparse stereo, followed by alignment between the current and previous point cloud [18], [26].

We adopt a solution based on incremental structure from motion with key frame selection and sparse local bundle adjustment as is more popular in the monocular case [11], [19], [25]. Given the additional step of feature matching between the left and right image, it is straightforward to apply to the binocular case. Instead of successive point cloud alignment, a robust image-based error model is used. Thus, no uncertainty modeling is required for the 3D landmarks and information over several key frames is easily combined. Relying on key frame selection allows real-time computation and reduces error accumulation. In some approaches, lowering the acquisition speed improves the results by preventing error accumulation [18]. In our approach, increasing the frame rate of the camera improves the accuracy of the trajectory because it does not increase the number of key frames but makes tracking faster and more reliable.

In the context of our work, relying on a stereo camera rather than on a single camera has important advantages. The first one is that newly observed landmarks can be immediately used for pose estimation in the next key frame. Secondly, this prevents any drift in the scale of the trajectory, a typical problem of monocular systems [32]. More importantly, the VO can fail and be re-initialized without requiring any information from the EKF.

These are the main steps of VO algorithm:

- Initialization:
  1) Feature detection in left and right images
  2) Sparse stereo matching
  3) Feature triangulation
- For each new image pair*:
  1) Feature detection in left and right images
  2) Feature matching between previous and current left images
  3) Feature matching between previous and current right images using constraints from the left image
  4) Sparse stereo matching on the remaining features
  5) Pose estimation using the 3-point algorithm
  6) Local bundle adjustment on the last b key frames

7) Relative pose computation and uncertainty estimation
8) Pose update to the EKF

* If any step fails, send pose update of infinite uncertainty and start from the beginning.

A detailed description follows.

A. Initialization

As is commonly done, we define our VO coordinate system by setting the initial pose of the stereo camera to be in canonical position and orientation. In practice, we can choose any coordinate system as long as we know the fixed rotation between the stereo camera and the IMU. Sparse stereo is then performed (more details below) and a first set of visual landmarks is computed by triangulation. If the features don't have enough disparity, we reject this frame and repeat the operation until initialization is reliable.

B. Feature detection, sparse stereo and tracking

We rely on the Harris corner detector with sub-pixel accuracy [16] and perform matching using Sum of Absolute Differences (SAD) on a eleven by eleven window. Pixels are processed by groups of 16, using Intel SSE4 instructions. We tested other more sophisticated detectors and descriptors [2], [3] but did not get substantial improvement on our datasets because they already contain a lot of distinctive features and were taken at a high frame rate (30 frames per second). Both tracking and stereo are performed by matching features while enforcing mutual consistency similarly to Nister *et al.* [25]. While being efficient, this approach also doesn't require any threshold for determining whether a match is valid or not. As previously mentioned, dense stereo is not performed.

In practice, pose estimation is only performed on a subset of the frames, called key frames. Thus, steps 1 to 3 are repeated until features show enough motion in the image. Matching between consecutive frames is done in two steps. First, we perform matching for the left images by restricting the search regions. We use fixed search regions because predicting feature position in the next frame is unreliable when the vehicle undergoes strong vibration. Matching of the right images is done similarly except that it can be sped up by intersecting the search region with epipolar lines provided by the matches in the left image. A common problem with corner detectors is that some features are not *stable* resulting in 'flickering' of their detection. To compensate for this weakness, features are not discarded as soon as matching fails, but only if it fails with the first subsequent key frame. For example, even if a feature from frame 1 does not appear in frame 2, matching may still be successful as long it appears in the following key frame and did not move outside of the search region. In practice, doing so is quite important in natural environments, especially if the number of frames between each key frame is large.

C. Pose estimation and local refinement

After tracking, we are given a set of n correspondences between image features and 3D landmarks, which we use to estimate the position of the stereo camera. We define the following quantities:

- k is the current key frame;
- $\boldsymbol{\Psi}_k$ contains the three Euler angles of the orientation and $\boldsymbol{R}_k$ is the position of the stereo camera center;
- We assume that the left and right camera have identical orientation and that their relative position to the camera center is given by $\boldsymbol{R}_l$ and $\boldsymbol{R}_r$;
- The 3D position in homogeneous coordinate of the observed landmarks are the rows of $\mathsf{s} \in \mathbb{R}^{4\times n}$;
- Corresponding measurements in the left and right images are the rows of $\mathsf{m}_{lk} \in \mathbb{R}^{2\times n}$ and $\mathsf{m}_{rk} \in \mathbb{R}^{2\times n}$, respectively.

Now, we can formally describe problem of the pose estimation at key frame k as

$$\arg\min_{\boldsymbol{R}_k,\boldsymbol{\Psi}_k} G(\boldsymbol{R}_k+\boldsymbol{R}_l,\boldsymbol{\Psi}_k,\mathsf{m}_{lk},\mathsf{s}_k)+G(\boldsymbol{R}_k+\boldsymbol{R}_r,\boldsymbol{\Psi}_k,\mathsf{m}_{rk},\mathsf{s}_k)$$

where G is the robust sum of squared re-projection error

$$G(\boldsymbol{R},\boldsymbol{\Psi},\mathsf{m},\mathsf{s}) = \sum_i \gamma\left(\boldsymbol{m}^i - \operatorname{proj}\left(g(\boldsymbol{\Psi})[\mathsf{I}_3|-\boldsymbol{R}]\boldsymbol{s}^i\right)\right),$$

g converts Euler angles to 3×3 rotation matrix, $\operatorname{proj}(\boldsymbol{X}) = \left[X_1/X_3, X_2/X_3\right]^\top$ is the projection function, γ is the truncated function:

$$\gamma(\boldsymbol{a}) = \begin{cases} \boldsymbol{a}^\top\boldsymbol{a} & \text{if } \boldsymbol{a}^\top\boldsymbol{a} < \varepsilon^2 \\ \varepsilon^2 & \text{otherwise} \end{cases}$$

and ε is the error threshold set to 1 pixel. An initial estimate is obtained by random sampling groups of three features from the left image and corresponding landmarks [13], [15], [25], followed by iterative refinement using the Levenberg-Marquardt algorithm.

Finally, we use sparse local bundle adjustment to simultaneously refine the pose estimate of the last b key frames (we use $b = 5$) as well as all the landmarks appearing in at least one of them [10]. To simplify the notation, let us define the current set of landmarks as s and assume they were all observed in the w previous key frames. Local bundle adjustment amounts to finding

$$\min_{\mathsf{s},\boldsymbol{R}_{k'},\boldsymbol{\Psi}_{k'} \text{ with } k'\in[k-b+1,k]} \sum_{k'=k-w}^{k} G(\boldsymbol{R}_{k'}+\boldsymbol{R}_l,\boldsymbol{\Psi}_{k'},\mathsf{m}_{lk'},\mathsf{s}) + G(\boldsymbol{R}_{k'}+\boldsymbol{R}_r,\boldsymbol{\Psi}_{k'},\mathsf{m}_{rk'},\mathsf{s})$$

Bundle adjustment is known to significantly improve the accuracy of the trajectory [19] and it significantly reduced the heading drift in our experiments.

Once bundle adjustment is done, it is straightforward to compute the position and orientation update of the current key frame with respect to the previous.

D. Uncertainty estimation

We found it essential to compute the pose update uncertainty. As illustrated in Figure 1 uncertainties tends to vary significantly along the trajectory. Computing the uncertainty is straightforward using a standard approach in non-linear parameter optimisation [4], [34]. We change our coordinate system so pose $k-1$ is located in canonical position and

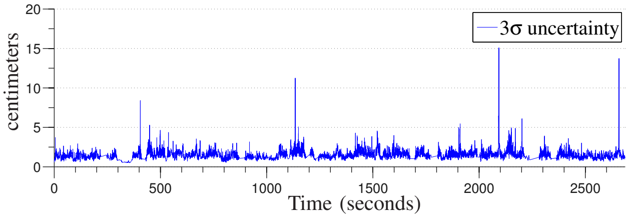


Fig. 1: Position uncertainty (in centimeter) of the motion update from the visual odometry for the Urban 2 dataset (see Figure 5)

orientation and transform the visual landmarks accordingly. We estimate the covariance matrix of the parameters by

$$\sigma^2 \left(\mathsf{J}^\top \mathsf{J}\right)^{-1}$$

where J is the jacobian of G evaluated at the current solution and σ is the standard deviation of the reprojection error of the currently observed landmarks. Note that we don't account for landmark uncertainty which we found to have little impact on the estimate.

V. Fusion

A. Algorithm Outline

We combine the inertial and VO data in an Extended Kalman Filter. A summary of the algorithm is given here with details in the following sections.

- Initialization:
  1) Determine initial alignment of IMU-camera frame to global navigation frame
  2) Calibrate IMU from static measurements
  3) Append estimates of initial position and orientation as delayed states
- Inertial Prediction:
  1) For each IMU measurement integrate state and uncertainty estimates
- Vision Measurement:
  1) For each VO measurement, form measurement prediction from current states
  2) Compute and apply state corrections using extended Kalman filter formulations with VO predicted measurement uncertainties if uncertainty is not infinite
  3) Reform state by appending updated estimates of current position and orientation as delayed states
  4) Return to Inertial Prediction step

B. Initialization and Inertial Prediction

The system state for filtering, $\boldsymbol{x} \in \mathbb{R}^{15\times 1}$, is defined by the current orientation, position and velocity of the IMU-camera sensor in global coordinates along with some IMU calibration parameters:

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{\Psi} & \delta\boldsymbol{\omega}^b & \boldsymbol{R}^n & \boldsymbol{V}^n & \delta\boldsymbol{f}^b \end{bmatrix}^\top \tag{1}$$

where $\boldsymbol{\Psi} = [\phi, \theta, \psi]^\top$ are the roll, pitch and heading Euler angles, relating the navigation frame (n) to the IMU-camera fixed body frame (b). $\delta\boldsymbol{\omega}^b$ are gyroscope biases in the IMU body frame, $\boldsymbol{R}^n$ is position in the navigation frame, $\boldsymbol{V}^n$ is velocity in the navigation frame and $\delta\boldsymbol{f}^b$ are the accelerometer biases in the IMU frame. The navigation frame is defined in this work as a North, East, Down (NED) relative coordinate system with origin at the initial location of the IMU-camera sensor. For clarity's sake, we leave out the notation for the fixed rotation between stereo camera and IMU. State prediction occurs via numerical integration of the state derivative with known initial conditions. Initial conditions are derived from: an alignment phase for roll and pitch angles, which uses the known gravity vector; a user input for heading, position and velocity; and a calibration phase for sensor biases. Additional sensors could be added to directly measure these initial conditions. State derivatives are kinematic relations between the states and IMU measurements with gravity and earth rotation rate as known inputs

$$\dot{\boldsymbol{x}} \quad = \begin{bmatrix} \mathsf{E}(\boldsymbol{\omega}^b - \delta\boldsymbol{\omega}^b - \boldsymbol{\Omega}^b + \boldsymbol{n}_{\boldsymbol{\omega}^b}) \\ \boldsymbol{n}_{\delta\boldsymbol{\omega}^b} \\ \boldsymbol{V}^n \\ \mathsf{C}_b^n(\boldsymbol{f}^b - \delta\boldsymbol{f}^b + \boldsymbol{n}_{\boldsymbol{f}^b}) - 2\boldsymbol{\Omega}^n \times \boldsymbol{V}^n + \boldsymbol{g}^n \\ \boldsymbol{n}_{\delta\boldsymbol{f}^b} \end{bmatrix}$$

where $\boldsymbol{f}^b, \boldsymbol{\omega}^b$ are the accelerometer and gyroscope measurements respectively, E is a matrix that relates Euler angle rates to gyroscope measured rotation rates, C_b^n is a rotation matrix, formed from the Euler angles, that relates the IMU-camera frame to the navigation frame, $\boldsymbol{g}^n$ is the known local gravity vector (incorporating centripetal acceleration terms) and $\boldsymbol{\Omega}$ is the known earth-rate vector (See [33] for details). Uncertainty is incorporated with Gaussian additive noises, $\boldsymbol{n}_*$ on $\boldsymbol{\omega}^b, \boldsymbol{f}^b, \delta\boldsymbol{\omega}^b$ and $\delta\boldsymbol{f}^b$, the first two of which represent true noise in the sensor, the last two represent modeling constraints. Uncertainty is defined by the covariance of the states

$$\mathsf{P} = E[(\boldsymbol{x} - \bar{\boldsymbol{x}})(\boldsymbol{x} - \bar{\boldsymbol{x}})^\top]$$

where E in the last expression represents the expected value operator, not to be confused with the Euler angle rate matrix E presented earlier. The covariance is propagated by numerical integration of the Lyapunov equation

$$\dot{\mathsf{P}} = \mathsf{F}\mathsf{P} + \mathsf{P}\mathsf{F}^\top + \mathsf{G}\mathsf{Q}\mathsf{G}^\top$$

where the following definitions apply

$$\mathsf{F} = \begin{bmatrix} \frac{\partial(\mathsf{E}(\boldsymbol{\omega}^b - \delta\boldsymbol{\omega}^b - \boldsymbol{\Omega}^b))}{\partial\boldsymbol{\Psi}} & -\mathsf{E} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathrm{I} & 0 \\ \frac{\partial(\mathsf{C}_b^n(\boldsymbol{f}^b - \delta\boldsymbol{f}^b))}{\partial\boldsymbol{\Psi}} & 0 & 0 & [-2\boldsymbol{\Omega}^n \times] & -\mathsf{C}_b^n \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix},$$

$$\mathsf{G} = \begin{bmatrix} \mathsf{E} & 0 & 0 & 0 \\ 0 & 0 & \mathrm{I} & 0 \\ 0 & 0 & 0 & 0 \\ 0 & \mathsf{C}_b^n & 0 & 0 \\ 0 & 0 & 0 & \mathrm{I} \end{bmatrix}.$$

These are simply the jacobians of the state derivative relative to state and noise respectively. Finally

$$\mathsf{Q} = \mathrm{diag}(E[\boldsymbol{n}_{\boldsymbol{\omega}^b}\boldsymbol{n}_{\boldsymbol{\omega}^b}^\top], E[\boldsymbol{n}_{\boldsymbol{f}^b}\boldsymbol{n}_{\boldsymbol{f}^b}^\top], E[\boldsymbol{n}_{\delta\boldsymbol{\omega}^b}\boldsymbol{n}_{\delta\boldsymbol{\omega}^b}^\top], E[\boldsymbol{n}_{\delta\boldsymbol{f}^b}\boldsymbol{n}_{\delta\boldsymbol{f}^b}^\top]).$$

For clarity the time dependence of these parameters is not explicit in the notation but it is seen that all except Q contain time varying values.

C. Delayed State Filter

At initialization, $k = 0$, the orientation and position states of the filter are copied such that a new state, $\hat{\boldsymbol{x}} \in \mathbb{R}^{21\times 1}$, is created by appending some delayed states $\boldsymbol{x}^d$ to the original states

$$\hat{\boldsymbol{x}} = \begin{bmatrix} \boldsymbol{x} & \underbrace{\boldsymbol{\Psi} \quad \boldsymbol{R}^n}_{\boldsymbol{x}^d} \end{bmatrix}^\top$$

where $\boldsymbol{x}$ is defined in Equation 1. The state covariance is similarly obtained

$$\hat{\mathsf{P}} = \mathsf{T}\mathsf{P}\mathsf{T}^\top = \begin{bmatrix} \mathsf{P}^{oo} & \mathsf{P}^{od} \\ \mathsf{P}^{od\top} & \mathsf{P}^{dd} \end{bmatrix}$$

where, we refer to states from the original state vector with superscript o and delayed states with superscript d. Here

$$\mathsf{T} = \begin{bmatrix} & \ddots & & & \\ & & \mathrm{I} & & \\ & & & \ddots & \\ \mathrm{I} & 0 & 0 & 0 & 0 \\ 0 & 0 & \mathrm{I} & 0 & 0 \end{bmatrix}$$

is a 21×15 matrix designed to select the appropriate rows of orientation and position covariance for copying. The result is an identical duplication of position and orientation estimates and their associated variances and covariances *at that instant in time*. The original states evolve with new measurements, while the delayed states remain fixed, in effect saving the information available at the instant they were appended. That is

$$\dot{\hat{\boldsymbol{x}}} = \begin{bmatrix} \mathsf{E}(\boldsymbol{\omega}^b - \boldsymbol{\delta\omega}^b + \boldsymbol{n}_{\boldsymbol{\omega}^b}) \\ \boldsymbol{n}_{\delta\boldsymbol{\omega}^b} \\ \boldsymbol{V}^n \\ \mathsf{C}_b^n(\boldsymbol{f}^b - \boldsymbol{\delta f}^b + \boldsymbol{n}_{\boldsymbol{f}^b}) - 2\boldsymbol{\Omega}^n \times \boldsymbol{V}^n + \boldsymbol{g}^n \\ \boldsymbol{n}_{\delta\boldsymbol{f}^b} \\ \boldsymbol{0} \\ \boldsymbol{0} \end{bmatrix}.$$

Similar formulations have been referred to in the literature as stochastic cloning [22]. The variances of the appended states are fixed (block diagonal elements) representing the uncertainty in the states at the instant they were appended. However the covariances (block off-diagonal) with the current state propagate as the current state continues to evolve

$$\dot{\hat{\mathsf{P}}} = \begin{bmatrix} \mathsf{F}\mathsf{P}^{oo} + \mathsf{P}^{oo}\mathsf{F}^\top + \mathsf{G}\mathsf{Q}\mathsf{G}^\top & \mathsf{F}\mathsf{P}^{od} \\ (\mathsf{F}\mathsf{P}^{od})^\top & 0 \end{bmatrix}$$

where the previous definitions of F, G and Q still apply.

D. Vision Measurement

At key frame k the state $\hat{\boldsymbol{x}}_k$ is given by

$$\hat{\boldsymbol{x}}_k = \begin{bmatrix} \boldsymbol{x}_k & \boldsymbol{\Psi}_{k-1} & \boldsymbol{R}_{k-1}^n \end{bmatrix}^\top. \tag{2}$$

Vision measurements provide change in position and change in orientation to the filter. Given the delayed state formulation, this measurement is formulated as a simple difference of the current and delayed states. The VO measurements are local and occur relative to the previous IMU-camera coordinates and not the global navigation frame. Generalizing the notation we have

$$\boldsymbol{z}_k = h(\boldsymbol{x}_k) + \boldsymbol{v} = \begin{bmatrix} g(\mathsf{C}_{b_k}^{b_{k-1}}) \\ \mathsf{C}_n^{b_{k-1}}(\boldsymbol{R}_k^n - \boldsymbol{R}_{k-1}^n) \end{bmatrix}$$

where $\mathsf{C}_{b_k}^{b_{k-1}} = \mathsf{C}_n^{b_{k-1}}\mathsf{C}_{b_k}^n$. The b_k notation indicates the orientation of the IMU-camera axes at time k and g converts a rotation matrix to Euler angles [33]. A direct subtraction of Euler angles will only yield a valid representation if the angle is small. The conversion to and from rotation matrices allows for large rotations to occur between key frames $k-1$ and k. The measurement function h is linearized with respect to the states $\boldsymbol{x}$ and a convential Kalman filter update is performed, using a Cholesky decomposition for numerical stability. Measurement uncertainty is represented by the covariance matrix of the noise $\boldsymbol{v}$ which is an output of the VO system as described in Section IV-D. Immediately after the update occurs, $\hat{\boldsymbol{x}}$ and $\hat{\mathsf{P}}$ are reformed so that the original and delayed states are briefly identical again before inertial prediction continues the cycle.

E. Measurement Noise Modeling

There is no guarantee that the measurement noise $\boldsymbol{v}$ is normally distributed or zero mean given the complexity of the VO algorithm. Without these guarantees the Kalman filter is suboptimal at best and potentially divergent. We can, however, use the central limit theorem to our advantage by accumulating VO measurements before incorporating them in the filter. In this manner, the total error will tend to satisfy the Kalman Filter assumptions for a sufficiently large accumulation time. Vision meaurements occur at an average rate of 12 Hz across our three datasets. We have found that accumulating 10 measurements before fusing in the filter works well. For brevity we omit the equations for this accumulation but the general form of the measurement equation still holds.

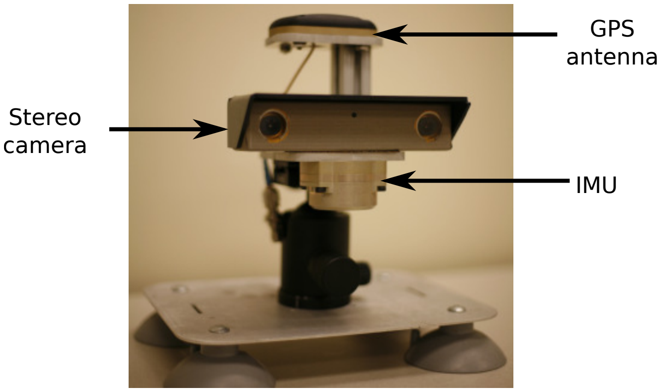


Fig. 2: Our navigation device. We treat the stereo camera and IMU as collocated with a fixed and known rotation.

F. Scale Estimation

In the urban datasets it is necessary to correct for small scale errors in the visual odometry measurements (see discussion in Section VI). This can be done in a principled manner by augmenting the state vector with a Gauss-Markov process [12]. The form of the measurement equation is then

$$\boldsymbol{z}_k = h(\boldsymbol{x}_k) + \boldsymbol{v} = \begin{bmatrix} g(\mathsf{C}_{b_k}^{b_{k-1}}) \\ s\mathsf{C}_n^{b_{k-1}}(\boldsymbol{R}_k^n - \boldsymbol{R}_{k-1}^n) \end{bmatrix}$$

where s is a single scale parameter that scales the three axis position measurement. In practice this parameter is normally a few percent in urban environments and is initialized with a small uncertainty to cover this range. We do not offer a formal proof but experiments show it to be observable and convergent in our datasets.

VI. Experimental validation

A. Equipment

The experimental system is based on commercial off the shelf hardware. A Point-Grey Research Bumblebee 2 stereo camera acquires rectified 640×480 stereo images at up to 30Hz, with a baseline of 12 cm and a field of view of 65 degrees. This baseline is ideal for ranges of a few meters which is sufficient in our agricultural application. For urban environments, a baseline of 30 cm would undoubtedly give better results. The inertial measurements come from a Honeywell HG1930 IMU running at 600Hz. Ground truth data is provided using tools from Novatel, Inc. including an OEMV-3 GPS receiver, Antcom 4G1521 antenna and the Waypoint post-processing software suite. The sensors are mounted on a specially designed mechanical mount providing alignment and positioning as well as versatile attachment possibilities (see Figure 2). In order to keep mounting options flexible, non-sensing equipment (including computing, power supply and GPS receiver) are stored in a box securely attached to the test vehicle at a convenient location.

B. Experiments

We conducted experiments on three datasets. Data were acquired and processed off-line in real-time. In the case of the urban datasets, we augmented the EKF with a global scale estimate for the VO update (Section V-F). Although initial orientation of the navigation device can be estimated using the IMU, this is only accurate to around two degrees. Thus to reduce this effect, we refine our initial orientation estimate by aligning the beginning of the estimated trajectories with the ground truth GPS. In practice, we must wait until the vehicle has turned at least one corner.

Our results are summarised in Table I and described in detail below. We compare our results with differential GPS which is accurate to a few centimeters. Figures 3, 4 and 5 show typical images from the corresponding datasets, a top view comparison of the trajectories and, finally, a comparison of the altitude. In the figures, starting points and ending points of the trajectories are marked with a purple X and O, respectively. Estimates of ground truth heading using GPS are only accurate to a few degrees, which is not sufficient for an accurate analysis. Visual inspection of the trajectories and altitude estimates suggests that the EKF provides some improvements to the heading, but clear improvements to the roll and pitch.

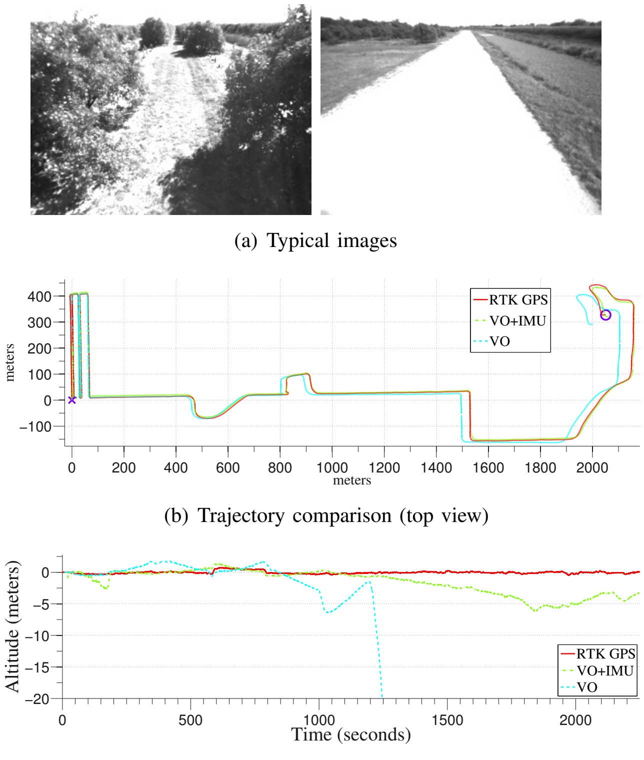
The first dataset (*Orange grove*, Figure 3) was obtained in Florida during February 2010 on the site of an orange grove using a commercial tractor. In the first half of the dataset, the tractor navigates in the rows of the orange grove at a speed between 5 and 10 km/h, which is the maximal speed for this type of application. In the second half, the tractor returnes to its garage at a speed of around 18 km/h. In these conditions, the tractor was subject to a lot of vibration and sudden changes in orientation (especially in the first half of the trajectory). This rendered feature tracking less reliable and affected the estimate of the the pitch and roll of the vehicle based on VO (see Figure 3c). The EKF could successfully correct for this problem yielding an altitude error of less than 5 meters for most of the trajectory. Heading drift was also surprisingly small resulting in well estimated straight lines of several hundred meters.

The second dataset (*Urban 1*, Figure 4) was acquired in the streets of Pittsburgh in January 2010 using a passenger car. It was taken in good conditions: at relatively low speed (at maximum of 33 km/h), in small streets with a lot of distinctive features, in a flat and quiet portion of the city with few other moving vehicles (see Figure 4a). As expected, the trajectory estimated by the VO shows little drift. Fusing with the IMU further improves heading as well as altitude estimates. Because of the small baseline of our stereo camera, we observed that speed was underestimated by around 3% (see traveled distance in Table I). However, we could correct this effect by augmenting the EKF with a scale correction, as described in Section V-F. With this addition, the traveled distance could be accurately estimated.

Our second urban dataset (*Urban 2*, figure 5) is a 20 km sequence also acquired in Pittsburgh. It is much more challenging than the first one as it was obtained in more realistic conditions. As seen in Figure 5a, several other vehicles would sometimes surround the car. Portions of the dataset were taken at high speed, up to 75 km/h. At that

Dataset	Time	Max. Speed	Dist.	VO+IMU 3D RMS	 Alt. RMS	 Dist./% err.	VO 3D RMS	 Alt. RMS	 Dist. /% err.
Orange grove	37 min.	15 km/h	5684	13.23	1.06	5695 / 0.19	120.73	116.92	5646 / 0.67
Urban 1	8 min.	33 km/h	2503	9.54	1.62	2502 / 0.05	13.12	2.54	2421 / 3.24
Urban 2	45 min.	75 km/h	20178	75.9	5.9	20007/ 0.86	113.61	33.24	19154 / 5.9

TABLE I: Comparison between vision aided-inertial and visual odometry for our three datasets. 'RMS' stands for Root Mean Square. All quantities are in meters unless otherwise stated. 'Time' stands for 'acquisition time', 'Dist' for 'traveled distance' and 'Alt.' for 'altitude'.



(a) Typical images

(b) Trajectory comparison (top view)

(c) Altitude comparison

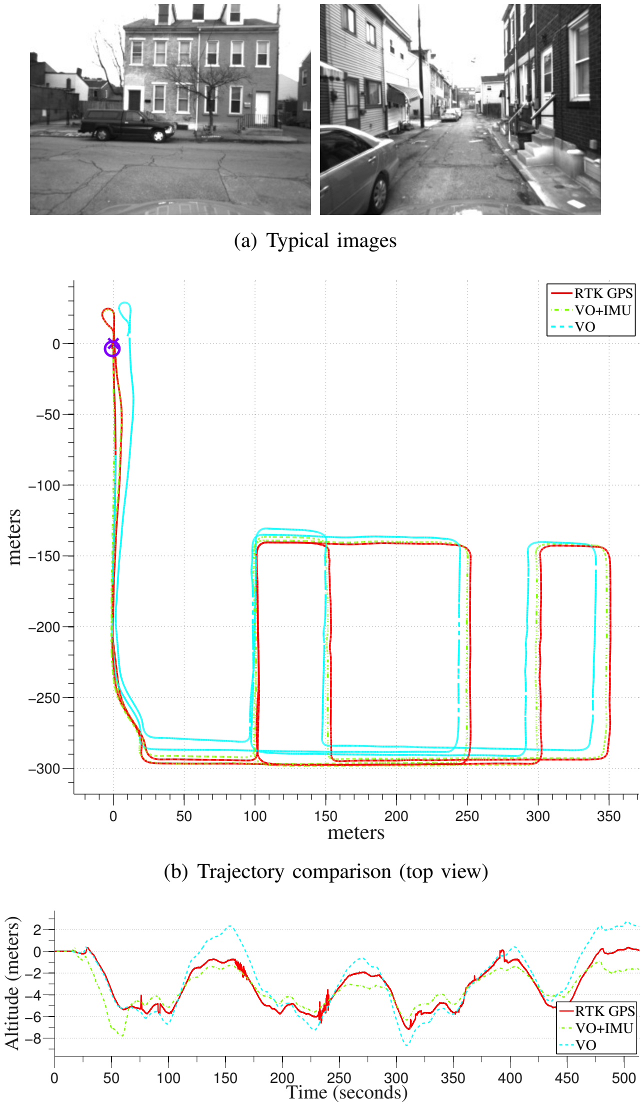
Fig. 3: Results for the Orange grove dataset. Maximum altitude error for the VO is around 400 meters.

speed, features located on the road were impossible to track even at 30 Hz. As a consequence, most landmarks were located far away from the stereo camera with a disparity close to zero, effectively reducing our stereo camera to a monocular system. In addition, the dataset was taken in a hilly portion of the city (see Figure 5c). Note that over that distance, a heading error of half a degree can result in close to a hundred meters in position error. Given the uncertainties of the VO updates (Figure 1), it seems that most of the drift is a result of inaccurate position updates from the VO. We are confident that a larger camera baseline would improve these results.

VII. Conclusion and Future Work

We presented a robust vision-aided inertial navigation system that can operate reliably in rural as well as urban environments. This system is still undergoing active development. One of the goals of this project is to provide centimeter-level accurate position estimates to an autonomous ground or aerial vehicle which, at present, typically rely on expensive RTK GPS. Our next step is the integration of other proprioceptive sensors, but most importantly conventional low-cost GPS.

We are also investigating the use of a single camera rather a stereo camera. This solution has several disadvantages but it would avoid the choice of baseline for the stereo camera,

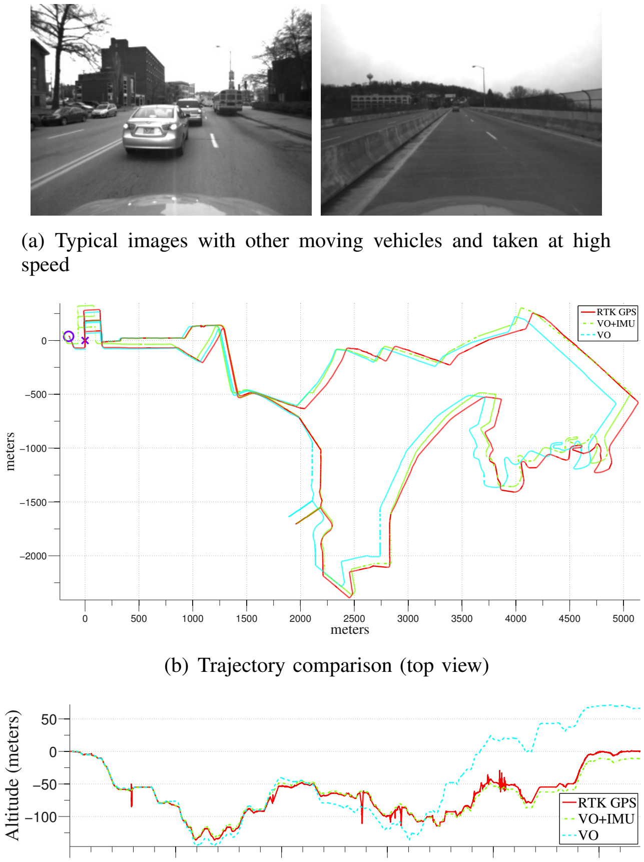


(a) Typical images

(b) Trajectory comparison (top view)

(c) Altitude comparison

Fig. 4: Results for the Urban 1 dataset.



(a) Typical images with other moving vehicles and taken at high speed

(b) Trajectory comparison (top view)

(c) Altitude comparison

Fig. 5: Results for the Urban 2 dataset.

perhaps making the system more flexible.

References

[1] M. Agrawal and K. Konolige. Rough terrain visual odometry. In *International Conference on Advanced Robotics*, 2007.

[2] M. Agrawal, K. Konolige, and M. R Blas. Censure: Center surround extremas for realtime feature detection and matching. *European Conference on Computer Vision*, 5305, 2008.

[3] H. Bay, T. Tuytelaars, and L. Van Gool. Surf: Speeded up robust features. *Lecture notes in computer science*, 3951:404, 2006.

[4] J. V Beck and K. J Arnold. *Parameter estimation in engineering and science*. Wiley New York, 1977.

[5] M. Bosse, W. C. Karl, D. Castanon, and P. DeBitetto. A vision augmented navigation system. In *IEEE Conference on Intelligent Transportation Systems*, page 10281033, 1997.

[6] J. Chen and A. Pinz. Structure and motion by fusion of inertial and vision-based tracking. *28th OAGM/AAPR Conference, Digital Imaging in Media and Education*, 179:5562, 2004.

[7] T. Cheviron, T. Hamel, R. Mahony, and G. Baldwin. Robust nonlinear fusion of inertial and visual data for position, velocity and attitude estimation of UAV. In *IEEE International Conference on Robotics and Automation*, page 20102016, 2007.

[8] A. J. Davison, I. D. Reid, N. Molton, and O. Stasse. MonoSLAM: Real-Time single camera SLAM. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6):1052–1067, 2007.

[9] David D. Diel, Paul DeBitetto, and Seth Teller. Epipolar constraints for Vision-Aided inertial navigation. In *IEEE Workshop on Motion and Video Computing*, volume 2, pages 221–228, Los Alamitos, CA, USA, 2005. IEEE Computer Society.

[10] C. Engels, H. Stewenius, and D. Nister. Bundle adjustment rules. *Photogrammetric Computer Vision*, 2, 2006.

[11] A. Eudes and M. Lhuillier. Error propagations for local bundle adjustment. In *IEEE International Conference on Computer Vision And Pattern Recognition*, 2008.

[12] J. Farrell. *Aided Navigation: GPS with High Rate Sensors*. McGraw-Hill, New York, 2008.

[13] 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):381–395, 1981.

[14] R. Garca-Garca, M. A Sotelo, I. Parra, D. Fernndez, J. E Naranjo, and M. Gaviln. 3D visual odometry for road vehicles. *Journal of Intelligent and Robotic Systems*, 51(1):113134, 2008.

[15] Haralick, Lee, Ottenberg, and Nolle. Review and analysis of solutions of the three point perspective pose estimation problem. *International Journal of Computer Vision*, 13(3):331–356, December 1994.

[16] C. Harris and M. Stephens. A combined corner and edge detector. In *Alvey vision conference*, volume 15, page 50, 1988.

[17] A. Hernandez-Gutierrez, J. I Nieto, T. Vidal-Calleja, and E. Nebot. Large scale visual odometry using stereo vision. *Australasian Conference on Robotics and Automation*, 2009.

[18] A. Howard. Real-time stereo visual odometry for autonomous ground vehicles. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, page 39463952, Nice, France, 2008.

[19] K. Konolige, M. Agrawal, and J. Sola. Large scale visual odometry for rough terrain. In *International Symposium on Robotics Research*, 2007.

[20] F. Lu and E. Milios. Globally consistent range scan alignment for environment mapping. *Autonomous Robots*, 4(4):333349, 1997.

[21] M. Maimone, Y. Cheng, and L. Matthies. Two years of visual odometry on the mars exploration rovers. *Journal of Field Robotics*, 24(3):169186, 2007.

[22] A. I Mourikis and S. I Roumeliotis. On the treatment of relative-pose measurements for mobile robot localization. In *IEEE International Conference on Robotics and Automation*, page 1519, 2006.

[23] A. I Mourikis and S. I Roumeliotis. A multi-state constraint kalman filter for vision-aided inertial navigation. In *IEEE International Conference on Robotics and Automation*, page 35653572, 2007.

[24] A. I Mourikis, N. Trawny, S. I Roumeliotis, A. Johnson, and L. Matthies. Vision aided inertial navigation for precise planetary landing: Analysis and experiments. In *Proc. Robotics Systems and Science Conference*, 2007.

[25] D. Nister, O. Naroditsky, and J. Bergen. Visual odometry for ground vehicle applications. *Journal of Field Robotics*, 23(1):3–20, 2006.

[26] C. F Olson, L. H Matthies, M. Schoppers, and M. W Maimone. Rover navigation using stereo ego-motion. *Robotics and Autonomous Systems*, 43(4):215229, 2003.

[27] L. L. Ong, M. Ridley, J. H Kim, E. Nettleton, and S. Sukkarieh. Six DoF decentralised SLAM. In *Australasian Conference on Robotics and Automation*, page 1016, 2003.

[28] G. Qian, R. Chellappa, and Q. Zheng. Robust structure from motion estimation using inertial data. *Journal of the Optical Society of America A*, 18(12):29822997, 2001.

[29] S. I Roumeliotis, A. E Johnson, and J. F Montgomery. Augmenting inertial navigation with image-based motion estimation. In *Proceedings-IEEE International Conference on Robotics and Automation*, volume 4, page 43264333, 2002.

[30] H. Strasdat, J. M. M. Montiel, and A. J Davison. Real-Time monocular SLAM: why filter? In *Int. Conf. on Robotics and Automation, Anckorage, AK*, 2010.

[31] D. Strelow and S. Singh. Motion estimation from image and inertial measurements. *The International Journal of Robotics Research*, 23(12):1157, 2004.

[32] J. Tardif, Y. Pavlidis, and K. Daniilidis. Monocular visual odometry in urban environments using an omnidirectional camera. In *International Conference on Intelligent Robots and Systems*, 2008.

[33] D. Titteron and J. Weston. *Strapdown Inertial Navigation Technology*. Institution of Electrical Engineers, Stevenage, UK, 2nd edition, 2004.

[34] B. Triggs, P. F McLauchlan, R. I Hartley, and A. W Fitzgibbon. Bundle adjustment-a modern synthesis. *Lecture Notes in Computer Science*, page 298372, 1999.

[35] M. Veth, R. C Anderson, F. Webber, and M. Nielsen. Tightly-Coupled INS, GPS, and imaging sensors for precision geolocation. *Air Force Institute of Technology,Department of Electrical and Computer Engineering*, 2008.