Hindawi Publishing Corporation
Journal of Robotics
Volume 2012, Article ID 634758, 15 pages
doi:10.1155/2012/634758

Research Article

Smart Localization Using a New Sensor Association Framework for Outdoor Augmented Reality Systems

F. Ababsa, I. Zendjebil, J.-Y. Didier, and M. Mallem

Laboratoire IBISC, EA 4526, Université d'Evry-Val-d'Essonne, 40 rue du Pelvoux, 91020 Evry, France

Correspondence should be addressed to F. Ababsa, ababsa@iup.univ-evry.fr

Received 17 February 2012; Revised 28 May 2012; Accepted 15 June 2012

Academic Editor: Huosheng Hu

Augmented Reality (AR) aims at enhancing our the real world, by adding fictitious elements that are not perceptible naturally such as: computer-generated images, virtual objects, texts, symbols, graphics, sounds, and smells. The quality of the real/virtual registration depends mainly on the accuracy of the 3D camera pose estimation. In this paper, we present an original real-time localization system for outdoor AR which combines three heterogeneous sensors: a camera, a GPS, and an inertial sensor. The proposed system is subdivided into two modules: the main module is vision based; it estimates the user's location using a markerless tracking method. When the visual tracking fails, the system switches automatically to the secondary localization module composed of the GPS and the inertial sensor.

1. Introduction

The idea of combining several kinds of sensors is not recent. The first multi-sensors system appeared with robotic applications where, for example, in [1] Vieville et al. proposed to combine a camera with an inertial sensor to automatically correct the path of an autonomous mobile robot. This idea has been exploited these last years by the community of Mixed Reality. Several works proposed to fuse vision and inertial data sensors, using a Kalman filter [2–6] or a particular filter [7, 8]. The strategy consists in merging all data from all sensors to localize the camera following a prediction/correction model. The data provided by inertial sensors (gyroscopes, magnetometers, etc.) are generally used to predict the 3D motion of the camera which is then adjusted and refined using the vision-based techniques. The Kalman filter is generally implemented to perform the data fusion. Kalman filter is a recursive filter that estimates the state of a linear dynamic system from a series of noisy measurements. Recursive estimation means that only the estimated state from the previous time step and the current measurement are needed to compute the estimate for the current state. So, no history of observations and/or estimates is required.

In [2] You et al. developed a hybrid sensor combining a vision system with three gyroscopes to estimate the orientation of the camera in an outdoor environment. Their visual tracking allows refining the obtained estimation. The system described by Ababsa [5] combines an edge-based tracking with inertial measurements (angular velocity, linear acceleration, magnetic fields). The visual tracking is used for accurate 3D localization while the inertial sensor compensates errors due to sudden motion and occlusion. The measurements of gravity and magnetic field are used to limit the drift problem. The gyroscope is employed to automatically reset the tracking process. Data provided by the two sensors are combined with an extended Kalman filter using a constant velocity model. More recently [9], the same authors proposed to use the GPS positions to re-initialize visual tracking when it fails. Thus, initialization of the visual tracking is obtained by defining a search area represented by an ellipse centered on the GPS position.

Recently, Bleser and Stricker [6] proposed to combine a texture-based tracking with an inertial sensor. The camera pose is predicted from data provided by the accelerometers using an Extended Kalman filter (EKF). In order to estimate the pose, the EKF fuse the 2D/3D correspondences obtained

from the image analysis and the inertial measurements acquired from the inertial sensor. A rendering of CAD model (textured patches) is made using the predicted poses. This allows aligning iteratively the textured patches in the current image to estimate the 2D motion and to update the estimate given by the filter. Natural feature points are tracked by a KLT (Kanade Lucas Tomasi) tracker. The motion model assumes constant acceleration and constant angular velocity. This approach needed offline preparation for generating a textured CAD model of the environment. Hu et al. [10] proposed to combine a camera, a GPS and an inertial gyroscope sensor. The fusion approach is based on PPM (Parameterized model matching algorithm). The road shape model is derived from the digital map with respect to GPS position, and matches with road features extracted from the real images. The fusion is based on a predictor-corrector control theory. After checking data integrity, GPS data will start a new loop and reset gyro's integrated. Gyro's prediction will be feedback into the gyro integration module as a dynamical correction factor. When the image feature tracking fails, gyro's prediction data is used for the camera pose estimation. Ababsa and Mallem [8] proposed a particle filter instead of the Kalman filter. Particle filters (PF), also known as methods of Monte-Carlo sequential, are sophisticated techniques for estimating models based on simulation. PFs are generally used to estimate Bayesian models. They represent an alternative to extended Kalman filter, their advantage is that they approach the optimal Bayesian estimation using enough samples. Ababsa et al. merged data from fiducial-based method with inertial data (gyros and accelerometers). Their fusion algorithm is based on a particle filter with sampling importance resampling (SIR). As the two sensors have different sampling frequency, the authors implemented two complementary filters. Thus, if there is no data of vision (e.g., occlusion), the system uses only data from the inertial sensor and vice versa. Aron et al. [11] used the inertial sensor to estimate the orientation of the camera only when the visual tracking fails. The orientation allows tracking the visual primitives by defining a search area in the image to perform the features matching. A homography is estimated from this set of matched features to estimate the camera pose. The errors of the inertial sensor are taken into account to optimize the search area. Unlike the approach proposed by Aron et al. [11] which only estimates the camera orientation, Maidi et al. [12] used an inertial sensor to estimate both the position and the orientation. Their multimodal system allows tracking fiducials and handling occlusions by combining several sensors and techniques depending on the existing conditions in the environment. When the target is partially occluded, the system uses a point-based tracking. In presence of a total occlusion of the fiducials, inertial sensor helps to overcome the vision failure. However, the estimation of position from acceleration produces drift over time resulting in a tracking failure. Combining sensors following the assistance scheme seems more interesting than the data fusion. Indeed, assistance approach makes the system more intelligent so that it can adapt itself to different situations and uses at each time only the data provided by the available sensors.

In this research work, we are interested in developing an original localization system combining three heterogeneous sensors (Camera, GPS and IMU) in order to increase the accuracy and the robustness. Our objective is to carry out a generic solution for the 3D localization, 3D visualization and interaction adaptable to several outdoor environments. The remainder of the paper is organized as follows: Section 2 is devoted to the overview of our Hybrid localization system. In Section 3, we give the formulation of the camera pose estimation problem when using point features. Sections 3 and 4, will describe in details our proposed hybrid localization system. Section 5 presents some performed experiments and results obtained in outdoor environments under real conditions. We conclude in Section 5 and suggest future works.

2. Hybrid Localization System Overview

Our localization system is wearable and composed of a Tablet PC, an Inertial Measurement Unit (IMU), a camera and a GPS receiver (see Figure 1). The IMU is rigidly coupled with the camera and used to estimate the camera rotation. We used an Xsens MTi sensor which contains gyroscopes, accelerometers and magnetometers. The advantage of the MTi is that it incorporates an internal digital signal processor which runs a real-time sensor fusion algorithm providing a reliable 3D orientation estimate. Data from MTi are synchronously measured at 100 Hz. For the vision, we opted for the uEye UI-2220-RE-C CCD camera with 6 mm lens which is extremely compact, low-cost and well adapted for outdoor environments. Color images with resolution of 768×576 pixels at a frame rate of 25 Hz are streamed to a PC using a USB 2.0 connection. A Trimble GPS Pathfinder ProXT receiver mounted on a user provides GPS measurements. The ProXT receiver integrates a multipath rejection technology providing a submeter accuracy. Its rate update is about 1 Hz. The ProXT receiver uses a Bluetooth wireless connection, to communicate with the computer. The three sensors providing measurements to the system are synchronized in hardware and runs at different rates.

This localization system uses an assistance scheme between the several sensors, and thus it is subdivided in two subsystems: a main subsystem and an auxiliary one. The main subsystem corresponds to the vision based localization. The auxiliary subsystem is used only when the visual tracking fails; it is composed of the GPS and the inertial sensors. This hybrid system estimates continuously the position and orientation of the point of view, even when the vision fails. Figure 2 provides a flow chart to describe the 3D localization process using the proposed assistance scheme.

3. Vision-Based Localization System

3.1. Camera Pose Problem Formulation. Throughout this paper, we assume a calibrated camera and a perspective projection model. If a point has coordinates $(x, y, z)^t$ in the coordinate frame of the camera, its projection onto the image plane is $(x/z, y/z, 1)^t$. In this section, we present the

Figure 1: Sensor components of our Hybrid tracker.

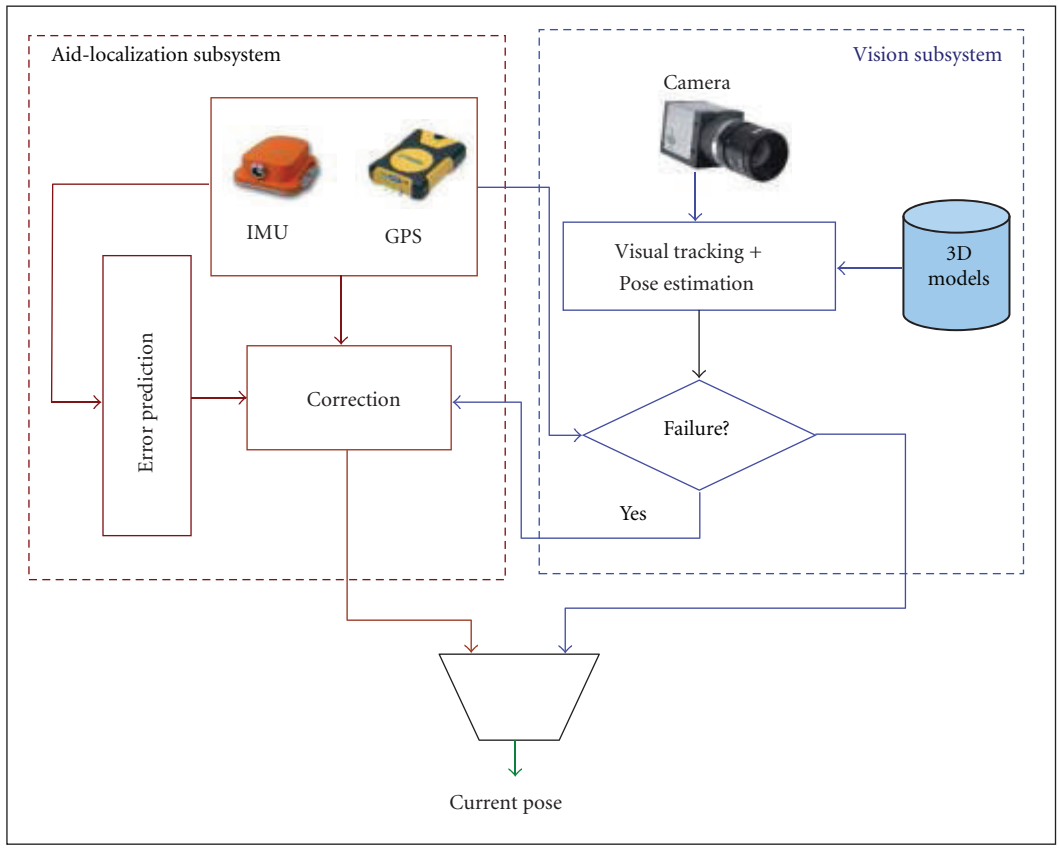


Figure 2: The system data flow scheme.

constraints for camera pose determination when using point and line features.

3.2. Problem Definition. Let $\mathbf{p}_i = (x_i, y_i, z_i)^t, i = 1, \ldots, n,\ n \geq 3$ a set of 3D non-collinear reference points defined in the world reference frame, the corresponding camera-space coordinates $\mathbf{q}_i = (x_i', y_i', z_i')$ are given by:

$$\mathbf{q}_i = R\mathbf{p}_i + T, \tag{1}$$

where $R = (\mathbf{r}_1^t, \mathbf{r}_2^t, \mathbf{r}_3^t)^t$ and $T = (t_x, t_y, t_z)^t$ are a rotation matrix and a translation vector, respectively. R and T describe the rigid body transformation from the world coordinate system to the camera coordinate system and are precisely the parameters associated with the camera pose problem.

Let the image point $\mathbf{g}_i = (u_i, v_i, 1)^t$ be the projection of $\mathbf{p}_i$ on the normalized image plane. Using the camera pinhole model, the relationship between $\mathbf{g}_i$ and $\mathbf{p}_i$ is given by:

$$\mathbf{g}_i = \frac{1}{\mathbf{r}_3^t \mathbf{p}_i + t_z}(R\mathbf{p}_i + T) \tag{2}$$

which is known as the colinearity equation.

The point constraint corresponds to the image space error, it gives a relationship between 3D reference points, their corresponding 2D extracted image points and the camera pose parameters as follows:

$$E_i^p = \sqrt{\left(\hat{u}_i - \frac{\mathbf{r}_1^t \mathbf{p}_i + t_x}{\mathbf{r}_3^t \mathbf{p}_i + t_z}\right)^2 + \left(\hat{v}_i - \frac{\mathbf{r}_2^t \mathbf{p}_i + t_y}{\mathbf{r}_3^t \mathbf{p}_i + t_z}\right)^2}, \quad (3)$$

where $\hat{\mathbf{m}}_i = (\hat{u}_i, \hat{v}_i, 1)^t$ are the observed image points.

The pose estimation problem is to find the rigid transform (R, t) that best fits the known 3D reference points with the observed 2D image points. Usually this is achieved by minimizing some form of accumulation of errors (least squares methods) based on (3). Typically Gauss-Newton or Levenberg-Marquardt methods are used for this purpose [13, 14].

3.3. Discussion. 3D-2D feature matching is critical for the camera pose estimation and still a difficult unsolved problem in computer vision. The tracking system needs an initialization that provides a set of good 3D-2D matched points. In practice, the accuracy of the matching process depends on the relevance of the information associated to the 3D points for their recognition. One interesting approach is to define a reference patches around the image points corresponding to the 3D model points. Matching is then performed by aligning these references patches within those extracted from the current frame. The correlation can be used to measure the similarity between the patches as in [15]. However, this method is not robust against illumination variation. Other approaches use the SIFT descriptors [16] for their robustness to changes in viewing conditions. The main disadvantage of the SIFT is its complexity and its high time consumption, this makes it not suitable for real-time applications. In this work, we propose two vision-based initialization approaches. The first one is semi-automated and requires the user intervention to guide the matching process; it is used to start the tracking system. The second approach is fully automated; it is executed when the tracking is lost. The next sections give an overview of these approaches.

3.4. Semi-Automated Initialization Approach. The proposed semi-automated initialization approach is performed in two steps: the wireframe model of the environment (here the building frontage) is first render, in real time, on the video flow coming from the camera, using a set of predefined poses (see Figure 3(a)). At the same time the user moves the camera in order to align the projected model within its image. Once this alignment is achieved (see Figure 3(b)) the user validates the corresponding pose and the system switch to the matching step in order to perform, with high accuracy, the 3D-2D points matching. Aligning the rendered model allows to limit the search area of the 2D points in the current image. This makes the approach faster and robust against the outliers.

The 3D-2D matching is performed as follows. A search box is defined around each projected 3D model point on the aligned image. The interest points are then extracted from these image regions. As 3D points represent corners in the model, we use the Harris detector [17] in order to extract the 2D interest points. Then, a SIFT descriptor is computed and associated to each extracted Harris point. We choose to use the SIFT descriptor because it is scale-rotation invariant and allows real-time tracking. The distances between the reference descriptor associated to the 3D point and the descriptors of the extracted 2D points are measured and compared. The 2D point that minimizes this distance is selected as the corresponding 2D point. We have also used a RANSAC algorithm [18] in order to detect and remove outliers in the matching set, and thus increasing the accuracy of the initialization.

In order to validate the whole matching 3D-2D points, we introduce a coherence test which is used as a quality measurement for the estimated camera pose. We assume that this pose is close to that selected when the wireframe model is aligned within the image. Let, $P_a = [R_a \mid T_a]$ be the predefined camera pose that is used for the model/image alignment, and $P = [R \mid T]$ the camera pose estimated using the set of candidate matched points. As the two matrices are identical, we can write then:

$$P_a \cdot P^{-1} = I, \quad (4)$$

where I is a 4×4 identity matrix.

So, the trace of the matrix $P_a \cdot P^{-1}$ tends to 4. Out coherence test can be formulated as:

$$\delta < \text{Trace}(P_a \cdot P) < 4, \quad (5)$$

where δ is a threshold below which the two matrices are considered different.

3.5. Automated Initialization Approach. Unlike the semi-automated approach described above, in automated initialization procedure, the user intervention is not required. The system switches automatically to this mode every time when the tracking fails because of noise, occlusion or image blurring. This approach is performed as follows (see Figure 4).

Let F_i be the reference frame that corresponds to the last captured image before the tracking failed. Several information are associated to this frame namely: the camera poses P_i and the set of matched 3D-2D points. The idea is to generate new 3D-2D matched points between the reference and the current frames.

For that, we first project the 3D points on the current frame using the reference camera pose P_i to generate predicted research areas. These image areas, named patches, are centred on the projected 3D points and have rectangular shape. The interest points corresponding to the SIFT features are then extracted inside these patches and matched with those extracted from the reference frame. We also use a RANSAC algorithm in order to discard the outliers. To find the 3D-2D matched points for the current frame, we only need to identify the transformation that maps interest points defined in the reference image to those extracted in the current image. We assume that this transformation is a

(a) (b)

FIGURE 3: (a) the model environment rendering. (b) manual alignment between the projected model and its image.

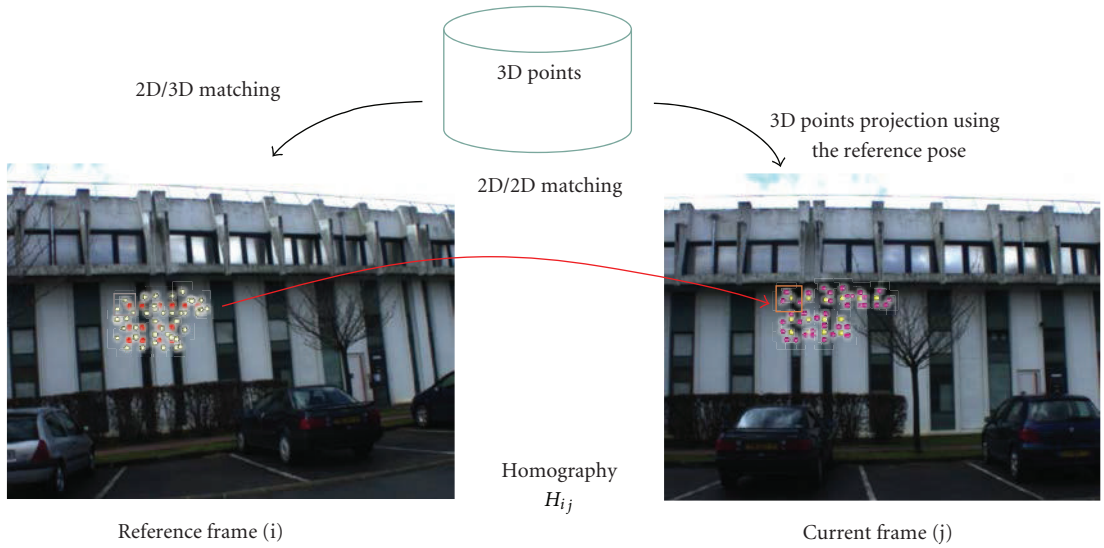


FIGURE 4: Automated initialization approach.

homography because the movement between the two frames is not meaningful.

Let H_{ij} be this homography, m_i and m_j the interest points extracted from reference and current frames F_i, and F_j, respectively. The relationship between m_i, m_j and H_{ij} is defined as:

$$m_j = H_{ij} \cdot m_i. \tag{6}$$

Once the homography is estimated, we apply it to the 2D image points associated to the 3D model points for the reference frame, in order to transform them in the current image. This allows updating correspondence between 3D and 2D points for the current image, and hence restarts automatically the tracking process.

3.6. Visual Tracking. Once the vision system is initialized the visual tracking can start. To estimate the camera pose, we must keep the 2D/3D matching for each current view. This can be achieved by using a frame-to-frame 2D points tracking. Tracking consists in following features from one frame t-1 to another frame t. Several approaches can be used such as correlation matching methods; however they are very expensive in computing time. To track 2D features in real time, the chosen method must be fast and accurate. For that the KLT Tracker [19] can be adopted. This algorithm used an optical flow computation to track features points or a set of predefined points from the previous image I_{t-1} to the current image I_t. Therefore, this algorithm tracks a set of 2D points associated to visible 3D points. Briefly, 2D points are searched in the neighborhood of its position in view t-1 based on the minimization of brightness difference. To minimize the computation time, the KLT tracker uses a pyramid of images for the current view. Therefore, tracking is done at the coarsest level and then propagate to the finest. This allows following the features over a long distance with great precision. The approach is fast and accurate, but it requires that the tracked points are always visible. So the approach does not handle occlusions.

4. IMU-GPS Localization System

This subsystem, replaces the vision-based localization system when this one fails. The position and orientation given by

the vision subsystem are substituted by the absolute position provided by the GPS receiver and the orientation given by the inertial sensor. The use of the Auxiliary subsystem is not limited only to replace the vision subsystem. The Auxiliary subsystem is also used to initialize the vision subsystem. Moreover, from the position and orientation given by this subsystem, we can measure the accuracy of the 3D localization estimated by the vision subsystem by defining some confidence intervals. The Auxiliary subsystem is composed of two modules: prediction and correction. The prediction module is used to predict accuracy errors of the localization system. It is based on online training of the error between the two subsystems. Once the localization system switches to the Auxiliary subsystem, the error is predicted following a Gaussian model and used to improve the position and the orientation provided by the GPS and the inertial sensor. The two parts composing the system interact continuously with each other. Also, the use of GPS for position estimation solves the problem of inertial sensor's drift, which is used only for orientation estimation.

4.1. Sensors Calibration. In our hybrid system, each sensor provides data in its own reference frame. The inertial sensor computes the orientation between a body reference frame attached to it and a local level reference frame. Also, the GPS position is expressed in an earth reference frame defined by WGS84 (World Geodetic System) standard. For registration, we need to estimate continually the camera pose which relates the world reference frame to the camera reference frame. Thus, the 3D localization provided by the IMU-GPS system must be aligned with the camera reference frame. The several sensors must be aligned in a unified reference frame in order to have the same position and orientation of the point of view. So, the hybrid sensor must be calibrated to determine the relationships between the several sensors and thus to unify the measurements. The accuracy of the IMU-GPS system depends on the accuracy of the calibration processes. We have developed an original calibration method in order to compute the several transformations with high accuracy. Our method is divided in two calibration processes. The first one consists in estimating the relationship between inertial sensor and camera (Inertial/Camera calibration). The second one estimates the transformation which maps the GPS position to the camera position (GPS/Camera transformation).

4.1.1. Inertial/Camera Calibration. In order to deduce the camera orientation from the orientation given by the inertial sensor, we need to estimate the transformation between the references frames attached to the camera and the inertial sensor. The used reference frames are illustrated in Figure 5.

Let R_{CW} be the rotation of the world frame R_W with respect to the camera frame R_C. R_{CI} represents the rotation of R_I with respect to the camera frame R_C. The rotation between the body frame R_I and R_G is noted R_{IG}. Finally, R_{GW} represents rotation of the world frame R_W with respect to R_G. The rotation R_{IG} is given by the inertial sensor and R_{CW} is obtained from the camera pose estimation. Thus, we need to compute rotation R_{CI} and R_{GW}. The relationship between the several frames is expressed by:

$$R_{CW} = R_{CI} \cdot R_{IG} \cdot R_{GW}. \tag{7}$$

In this case, the Inertial/Camera calibration process consists in estimating the rotation matrix R_{CI} and deducing the matrix R_{GW}. We assume that Z-axis of the R_G frame is pointing up along the vertical and is collinear with the Z-axis of the world coordinate frame R_W (Figure 6).

Therefore, this configuration implies that the matrix R_{GW} will correspond to a single rotation around the Z-axis with an angle θ. So, the matrix R_{GW} can be expressed as follows:

$$R_{GW} = \begin{pmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{8}$$

Assuming that $R_{IG} = (r_i^{IG})$, $i = 1, 2, 3$, where r_i^{IG} is the ith column of the R_{IG} matrix. Equation (8) can be written as:

$$R_{CW} = \left(R_{CI} \cdot r_1^{IG} \;\; R_{CI} \cdot r_2^{IG} \;\; R_{CI} \cdot r_3^{IG}\right) \cdot \begin{pmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{9}$$

and then,

$$R_{CW} = \left(R_{CI} \cdot r_1^{IG} \cdot \begin{pmatrix} \cos(\theta) \\ \sin(\theta) \end{pmatrix} R_{CI} \cdot r_2^{IG} \cdot \begin{pmatrix} -\sin(\theta) \\ \cos(\theta) \end{pmatrix} R_{CI} \cdot r_3^{IG}\right) \tag{10}$$

From (10) we can deduce that:

$$r_3^{CW} = R_{CI} \cdot r_3^{IG} \tag{11}$$

We rewrite this equation in the linear form $A \cdot X = b$, so:

$$\begin{pmatrix} \left(r_3^{IG}\right)^T & 0 & 0 \\ 0 & \left(r_3^{IG}\right)^T & 0 \\ 0 & 0 & \left(r_3^{IG}\right)^T \end{pmatrix} \cdot X = r_3^{CW}, \tag{12}$$

where $X = \left(r_{11}^{CI} \; r_{12}^{CI} \; r_{13}^{CI} \; r_{21}^{CI} \; r_{22}^{CI} \; r_{23}^{CI} \; r_{31}^{CI} \; r_{32}^{CI} \; r_{33}^{CI}\right)^T$. We need at least 3 matrixes R_{IG} and R_{CW} in order to estimate R_{CI} using the least mean squares algorithm. Once the matrix R_{CI} is computed we can deduce the matrix R_{GW} using (1), so:

$$R_{GW} = R_{IG}^T \cdot R_{CI}^T \cdot R_{CW}. \tag{13}$$

The estimation of the two matrixes R_{CI} and R_{GW}, which is done off line, allows the system to estimate the rotation of the camera in real-time using only the orientation matrix given by the inertial sensor. This is very important to recover the camera pose when the visual tracking fails.

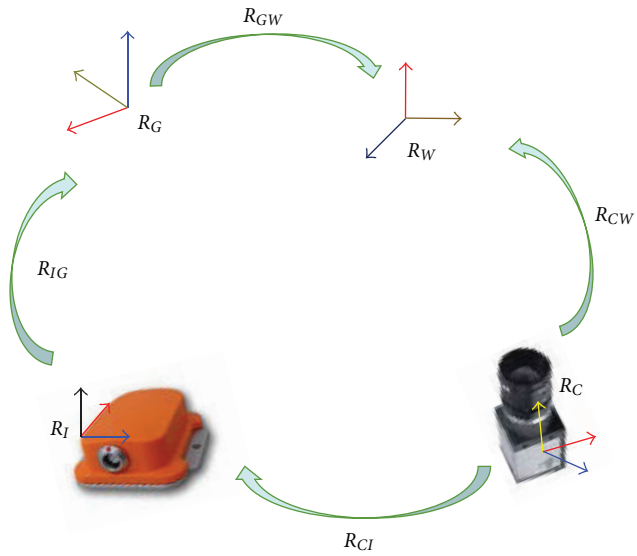


Figure 5: Inertial/Camera: Reference frames configuration.

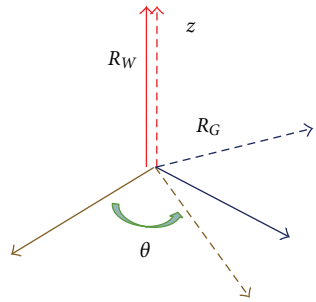


Figure 6: The relationship between the world reference frame R_W and the global inertial reference frame R_I.

4.1.2. GPS/Camera Calibration. This second calibration process estimates the rigid transformation (rotation + translation) which maps the GPS position to the local world reference frame R_W (see Figure 6). In our case, the position provided by the GPS receiver is expressed in degrees. A Lambert conic projection is necessary to express the GPS positions in meters. This projection minimizes the distortions and is adapted to the region where the experiments are done (i.e., France). The Lambert conic projection superimposes a cone over the sphere of the Earth. Let p_{gps} the camera position given by the GPS receiver and expressed in the earth reference frame and p_{cam} the camera position in the local world reference frame. The relationship between p_{gps} and p_{cam} is then given by:

$$p_{\text{cam}} = R \cdot p_{\text{gps}} + t, \tag{14}$$

where p_{cam} is computed using the camera calibration parameters as follows: $p_{\text{cam}} = -R_{CW}^{T} \cdot t_{CW}$.

Finally, the rigid transformation is obtained by minimizing the following criterion:

$$\sum_{i=1}^{n} \left\| p_{\text{cam}}^{i} - \left(R \cdot p_{\text{gps}}^{i} + t \right) \right\|^{2}. \tag{15}$$

In order to simplify the problem we have used the axis and the angle $(\vec{n}, \theta)$ to represent the rotation matrix R [20]. This allows subdividing the optimization problem into two sub-problems, one for the rotation estimation and one for the translation vector.

4.2. Localization Error Prediction. The estimation of the 3D localization provided by the combination of the GPS and the inertial sensor is less accurate then the vision-based estimation. The computation of the produced error is important in the localization process. Indeed, it allows quantifying the quality of measurements in order to improve the 3D localization estimation provided by the IMU-GPS localization system. The error represents the offset between the camera pose and the position and orientation deduced from GPS and inertial sensor. When the vision fails, this error must be predicted. For that, the error is modeled as a regression with a Gaussian process [21]. The Gaussian process is a stochastic process which generates samples and can be used as a prior probability distribution over functions in Bayesian inference.

Let be $x_1, x_2, \ldots, x_n$ a set of training data associated to $y_1, y_2, \ldots, y_n$ where $y_i = f(x_i)$. The goal is to predict

the value y_{n+1} associated to the data x_{n+1}. We consider $(Y_1, Y_2, \ldots, Y_{n+1})$ a set of $(n+1)$ random variables which have a Gaussian distribution with zero mean and covariance matrix Σ_{n+1} such as:

$$\Sigma_{n+1} = \begin{pmatrix} \Sigma_n & \kappa \\ \kappa^T & \kappa_{n+1} \end{pmatrix}, \tag{16}$$

Where κ_n is $n \times n$ matrix, κ is a n-column vector and κ_{n+1} is a scalar. If $y_1, y_2, \ldots, y_n$ are the observed variables associated to $x_1, x_2, \ldots, x_n$ then the conditional distribution $P(Y_{n+1} | Y_1, \ldots, Y_n)$ yielding a Gaussian distribution with:

$$\begin{aligned} \mu_{Y_{n+1}} &= \kappa^T \Sigma_n^{-1} y^n \\ \sigma^2_{Y_{n+1}} &= \kappa_{n+1} - \kappa^T \Sigma_n^{-1} \kappa, \end{aligned} \tag{17}$$

where $y^n = (y_1, y_2, \ldots, y_n)^T$, $\kappa_{n+1} = \mathrm{cov}(y_{n+1}, y_{n+1})$, $\kappa_i = \mathrm{cov}(y_{n+1}, y_i)$ and $\Sigma_{ij} = \mathrm{cov}(y_i, y_j)$. The covariance between y_i and y_j is the same as the covariance between x_i and x_j whish is given by:

$$\mathrm{cov}\left(x_i, x_j\right) = \frac{1}{N - |i - j|} \sum_{n=1}^{N-|i-j|} x_n \cdot x_{n+|i-j|}. \tag{18}$$

In our case x_i represents either the GPS position or the orientation matrix give by the inertial sensor, and y is the error localization that we want to predict. So, during the visual tracking, the offset between the IMU-GPS localization system and the vision-based localization system is recorded for the online training step. Training data are used to learn sampled covariance function. When the visual tracking fails, the Gaussian process predicts the offset made by GPS and the inertial sensor. This offset, which is represented by the mean error, is used to correct the estimated 3D localization. Indeed, the Gaussian Process allows computing $p(y \mid x)$ the likelihood of the error localization. This likelihood function is Gaussian, with mean and variance at the point x given by the Gaussian Process based map. In our case, mean and variance are used to compute the error ellipse of the estimated localization.

5. Real-Time Operating

Our localization system operates using a finite state machine scheme (see Figure 7). A finite state machine is an abstract model composed of a finite number of states, transitions between those states, and actions. This formalism is mainly used in the theory of computability and formal languages and allows running in real-time with high accuracy.

The proposed state machine is composed of three states: the Auxiliary predominance state, the initialization state and the visual predominance state. The transitions between different states are as follows: At the initialization state, the Auxiliary subsystem provides an estimation of the pose (1). This estimation is refined with vision subsystem (2). When the visual tracking fails, the Auxiliary subsystem takes over to estimate the 3D localization (3). Since the Auxiliary subsystem is less accurate than the vision subsystem, the estimation is corrected taking into account the predicted error. Thereafter, the estimation is used to re-initialize the visual tracking (4).

We have defined three criteria in order to quantify the quality of the estimated pose using the visual tracking. If one of these criteria is not verified, the pose is rejected and the system switches to the Auxiliary subsystem.

5.1. Number of Tracked Points. The number of 2D/3D matching points affects the accuracy of the minimization process used to estimate the camera pose. Indeed, the more we have a large set of 2D/3D matched points, the more the estimated pose is accurate and vice versa. For this, we define a minimum number of matching. Below this threshold, it is considered impossible to estimate the pose with the vision subsystem.

5.2. Projection Error. The number of matched points is not sufficient to ensure the accuracy of the pose estimation; the projection error criterion can also be used. This error represents the average square of the difference between the projection of 3D points using estimated pose and the 2D points. If the error is large, greater than an empirical threshold, the pose is considered wrong.

5.3. Confidence Intervals. The data provided by the Auxiliary subsystem can also be used as an indicator of the pose validation. In fact, from the position and orientation given by the Auxiliary subsystem, confidence intervals are defined. They are represented by an ellipsoid centered by the orientation provided by the inertial sensor and an ellipse which center is determined by the 2D position given by GPS. The axes of the ellipse or the ellipsoid can be defined $3 * \sigma$ (standard deviation of the offset between the camera pose and Auxiliary estimation) or empirically. If the pose computed by the vision subsystem is included in these confidence intervals (position in the ellipse and the orientation in ellipsoid), the pose is considered correct.

6. Experiments

Several experiments have been achieved to study the behavior of the proposed localization system when used in outdoor environments. The first experiment points out the performances of the semi-automated initialization approach in several conditions. Figure 8 shows that this approach performs good matching in spite of the illumination change.

Furthermore, in order to analyze the error in the matching process, we estimate the mean distance between the 2D points obtained after the semi-automated initialization step and the 2D points extracted from the images after refining the camera pose. We found a mean error equal to 3.8216 pixels with a standard deviation about 1.0873 pixels. This means that semi-automated approach is very efficient to generate a rough estimate of 2D-3D correspondences and really helps the tracking system to rapidly converge to the optimal solution.

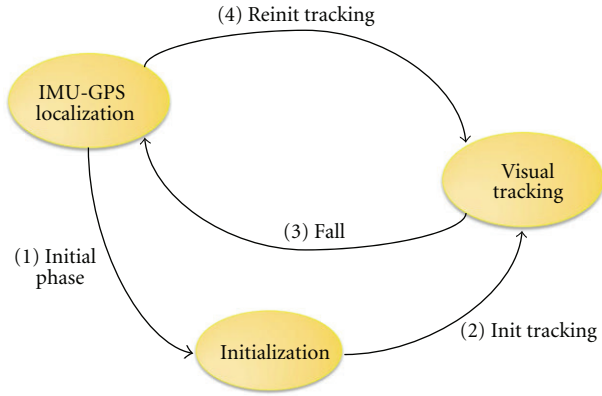
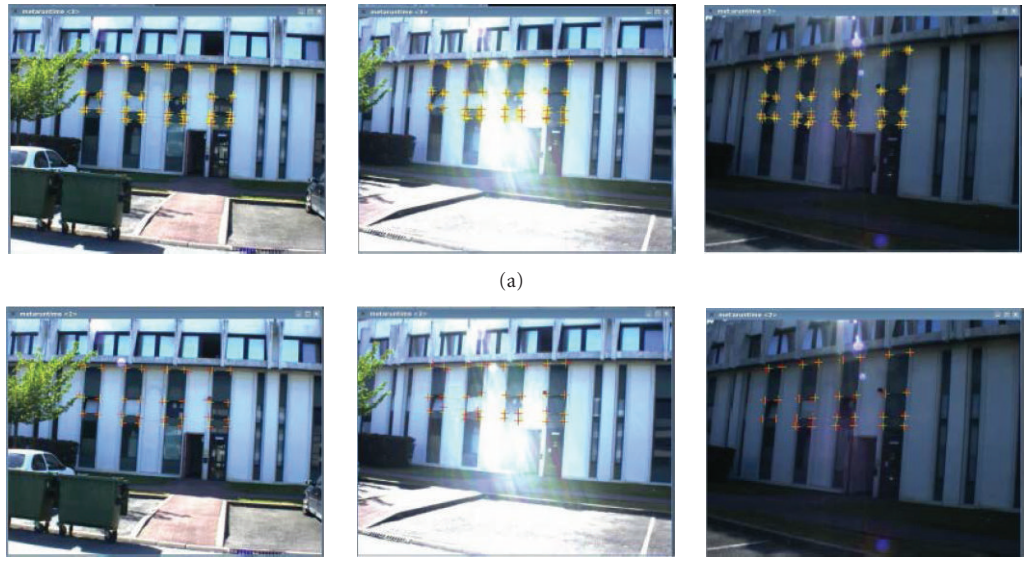


Figure 7: The state machine scheme of 3D localization system's operation.



(a)

(b)

Figure 8: Performance of semiautomated approach.

Table 1: Computation times for the semiautomated initialization approach.

Steps	Times
Manual alignment	unknown
Extraction and matching	50 ms
RANSAC	100 ms
Total (without manual alignment)	150 ms

We also analyzed the execution time by carefully evaluating the processing time needed to achieve each step in the semi-automated initialization procedure. An example of these computation times is given in Table 1.

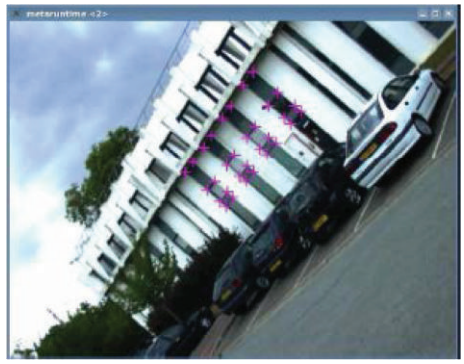
This table shows that the computation time needed to achieve the semi-automated initialization matching is quite fast and makes this approach particularly efficient for the initialization stage of the tracking system.

In addition, we also tested the performances of our automated initialization approach. For that we have considered several images taken under different viewpoints. Figure 9 shows some obtained results. We can see that, for all the considered cases, the reference points (taken on the frontage of the building) are well matched in the current frames. In this example we have considered coplanar points.

We have also tested our approach for non coplanar points chosen on the tower of the castle (Figure 10). Obtained results in this case are satisfying. Indeed, combining the SIFT points with the RANSAC algorithm provides an accurate and robust homography estimation, and thereby allows good points matching. The matching process in this case gives a mean error about 1.7823 pixels with a standard deviation of 0.6634 pixels.

The computation time of this approach depends mainly on the SIFT features extraction and matching. Introducing a prediction stage in order to limit the research area of the interest points in the current frame has significantly reduced

(a) Reference Image

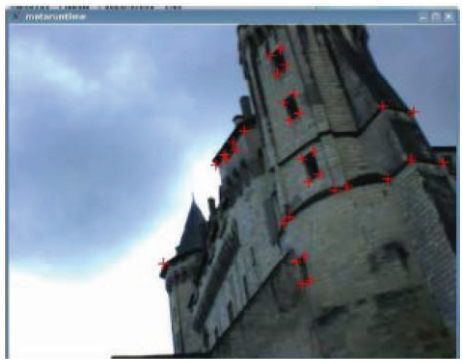


(b) 1st result

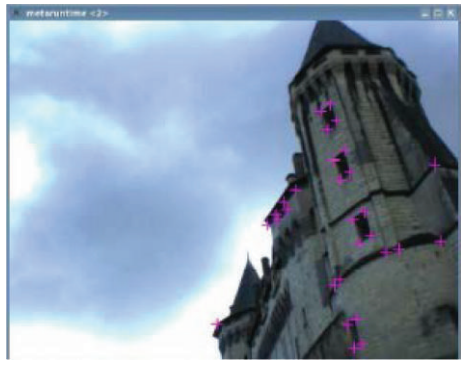
(c) 2nd result

(d) 3rd result

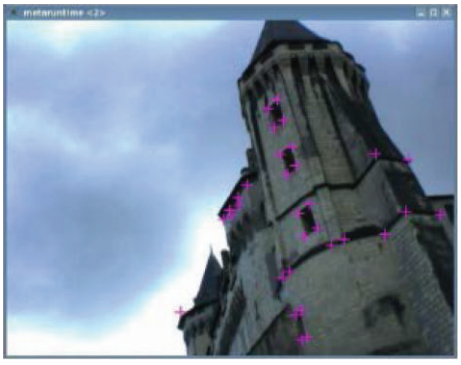
Figure 9: Matching results for autometed initialization approach, case of the coplanr points.



(a) Initial Image



(b) 1st Result



(c) 2nd Result

(d) 3rd Result

Figure 10: Matching results for autometed initialization approach-Non coplanr points.

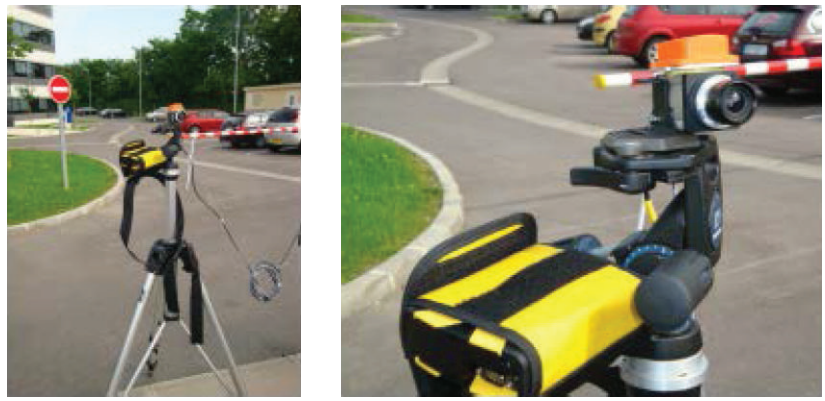


Figure 11: Our hybrid sensor mounted on a tripod.

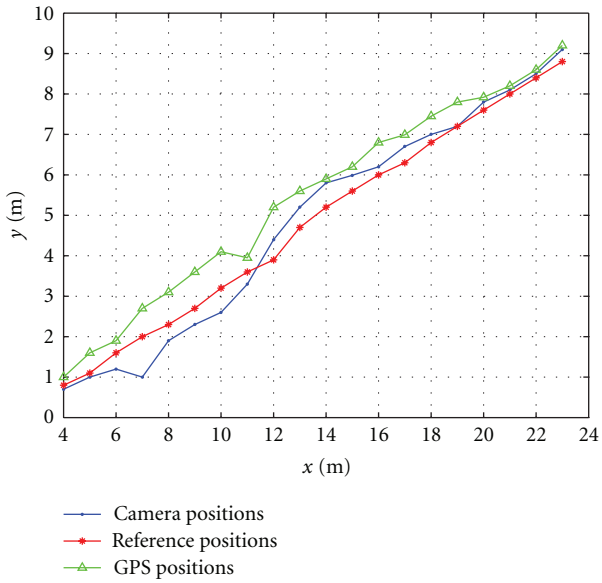


Figure 12: The straight line test: reference position versus camera position versus GPS position.

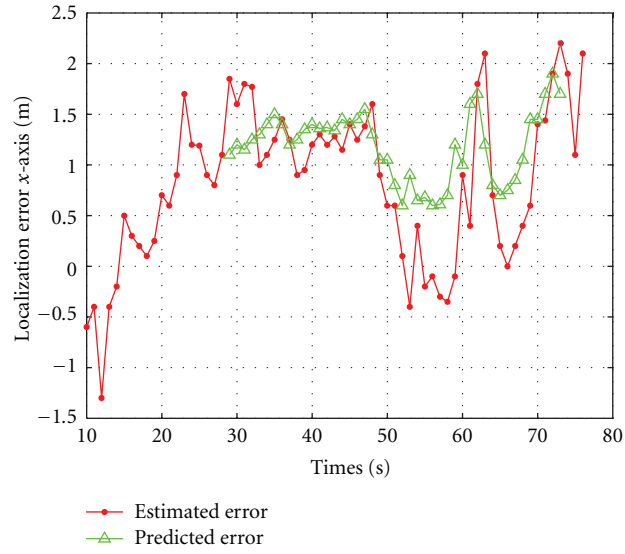


Figure 13: Predicted and estimated localization errors (on x-axis).

the total time of the whole algorithm (practically, it is divided by two). Comparing to other similar approaches [15], our proposed automated initialization technique is more flexible and provides best real times performances.

In the next experiments, we have evaluated the performance of our GPS-IMU localization system. The first experiment considers a straight line as a truth data. The origin of this line is defined in front of the origin of the world reference frame. The line is sampled in several positions and for each sample we do some acquisitions, namely images and GPS positions. The sensors are mounted on a tripod to ensure more stability (see Figure 11). The reference positions are measured with a telemeter which accuracy is about 0.15 m. In addition, for each acquired image, we calculated the position and the orientation of the camera.

From the GPS data and the transformation estimated during the calibration step, we deduce the absolute position with respect to the world reference frame associated to the real scene. By comparing the different estimated positions to the reference positions, we find a mean offset about (1.8374 m; 1.4810 m). The same GPS positions compared to the camera's positions give a mean error equal to (1.7321 m; 1.4702 m) with a standard deviation (1.8314 m; 1.0116 m). Figure 12 shows the hole trajectories obtained from the GPS and camera positions computations compared to the reference trajectory.

The second experiment focused on the relative position between two successive fixed positions. In average the offset between the reference position and that obtained with the GPS is about 0.7817 m with a standard deviation equal to 1.06 m. Similar values are given by the vision subsystem, that is, an offset mean about 0.8743 m with a standard deviation of 0.9524 m. Therefore, these results demonstrate that the movement provided by the two subsystems is consistent. The third experiment performed several continuous recordings of GPS/camera positions. The two sensors are time-stamped in order to synchronize them and to retrieve the set of data acquired at the same time. The positions given by the

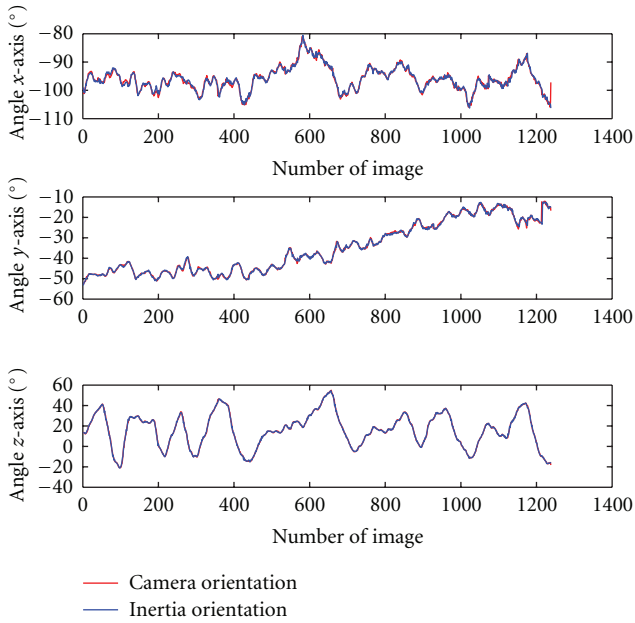


FIGURE 14: The camera's orientation versus inertial sensor's orientation.

vision and the GPS without correction are compared and the obtained errors are about 0.9235 m in the x-axis (with a standard deviation of 0.6669 m) and 0.8170 m in the y-axis (with a standard deviation of 0.6755 m). In addition, in order to study the error prediction approach we first used a set of 76 data acquired in continuous manner to perform the error training. Then, the Gaussian process is used with the last 30 data to predict errors. The mean offset between the predicted error and the real one is about ($\mu_x = 0.2742$ m; $\sigma_x = 0.4799$) and ($\mu_y = 0.5757$ m; $\sigma_y = 0.5097$ m) (see Figure 13). The positions provided by the GPS receiver are then corrected using this predicted error. This allows improving the 3D localization provided by the Auxiliary subsystem.

To assess the accuracy of the inertial sensor, we compared the orientations produced from the gyroscope to those computed by the vision pose estimation algorithm. For that, a video with several orientations in an outdoor environment has performed. Both orientations have the same behavior (Figure 14).

However, in some cases, we found that external factors can affect the inertial measurements, particularly in defining the local reference frame where the x-axis is in the direction of the local magnetic north. This causes errors in the orientation estimation. To solve this problem the rotation between the local reference frame associated to inertial sensor and the world reference frame is re-estimated continuously. The behavior of the whole system is also tested. The initialization process allows having the matching of the 3D visible points from the 3D model with their projections in the first view. From this 2D/3D matching, the set of 2D points are defined and tracked frame to frame. For each frame, the wire frame model is registered using the positions and orientations obtained from the hybrid localization system. In Figure 15, the green color projection is obtained from the positions and orientations provided by the vision subsystem. The wire frame model is well superimposed on the real view which demonstrates the accuracy of the camera pose estimation. In magenta, the projected model is obtained with the positions and orientations provided by the Auxiliary subsystem. Figure 15 show that when vision fails, the localization system switches to the Auxiliary subsystem to provide 3D localization. The localization is corrected with the predicted error which contributes to improve the estimation.

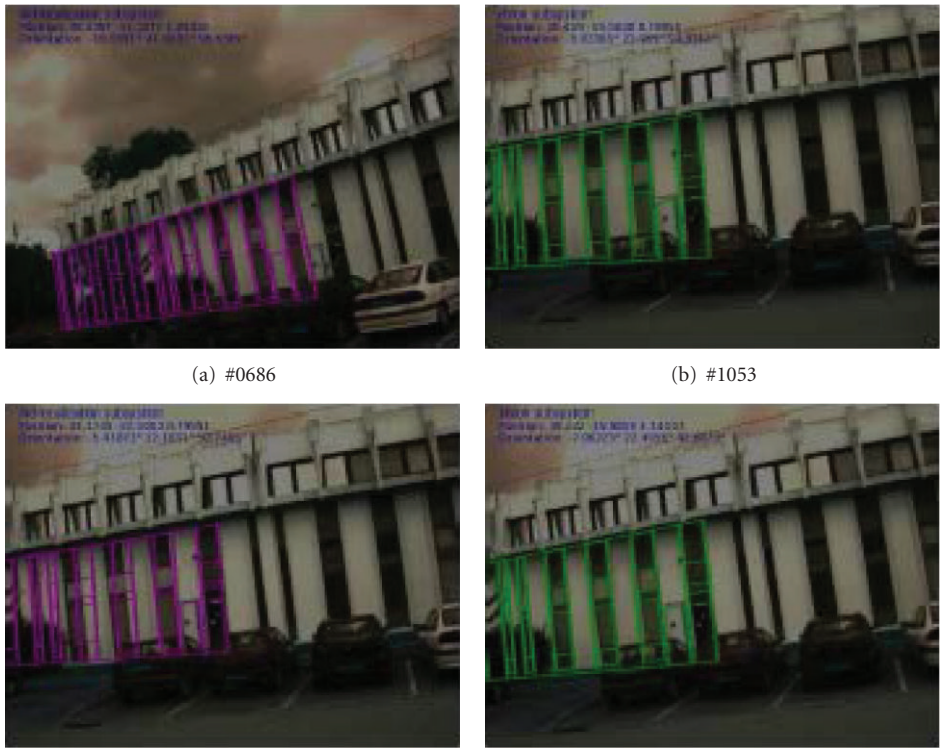
Figure 16 show that during the occlusion of the tracked points, the GPS-IMU localization system provides always an estimation of the position and orientation of the camera. Therefore, even when a total occlusion occurred, the system can provide a rough estimation of the 3D localization. This would not be the case if we used only the camera.

Furthermore, we evaluated the performance of our 3D localization system in this urban scene. The 3D model of the building is known, it is composed of 120 natural points defined by their 3D coordinates within the world coordinates frame. Several trials in different locations were recorded. The data coming from the three sensors are time-stamped and stored in data file. The camera pose estimated by the visual tracking was used as ground truth for the performance evaluation. We have defined a set of 2D/3D point's correspondences from the first sequence frame. The interest points are then tracked frame to frame using the KLT feature tracking algorithm [17, 18]. The 2D/3D correspondences are then updated and used to estimate the 3D localization. We have compared our approach with a data fusion method using an Extended Kalman Filter. Computational results show the effectiveness of our approach, since it obtains better accuracy. Figure 17 shows the recovered camera trajectory in world coordinate system; we note that the trajectory estimated by our localization approach is closer to the ground truth then the one estimated by fusing data coming from the three sensors.

Finally, the performances of our localization system are compared to the state-of-the-art results reported by DiVerdi and Höllerer [22] with their GroundCam system which also combines a camera, a GPS receiver and an orientation tracker. The authors run their system along a residential street for approximately 90 seconds and compare the estimated trajectory to a hand-labelled ground truth. The reported RMS is about 5.5 m. We run our system in the similar conditions around our institution building. The obtained RMS is about 1.55 m which shows that our localization system generates results significantly better.

7. Conclusion

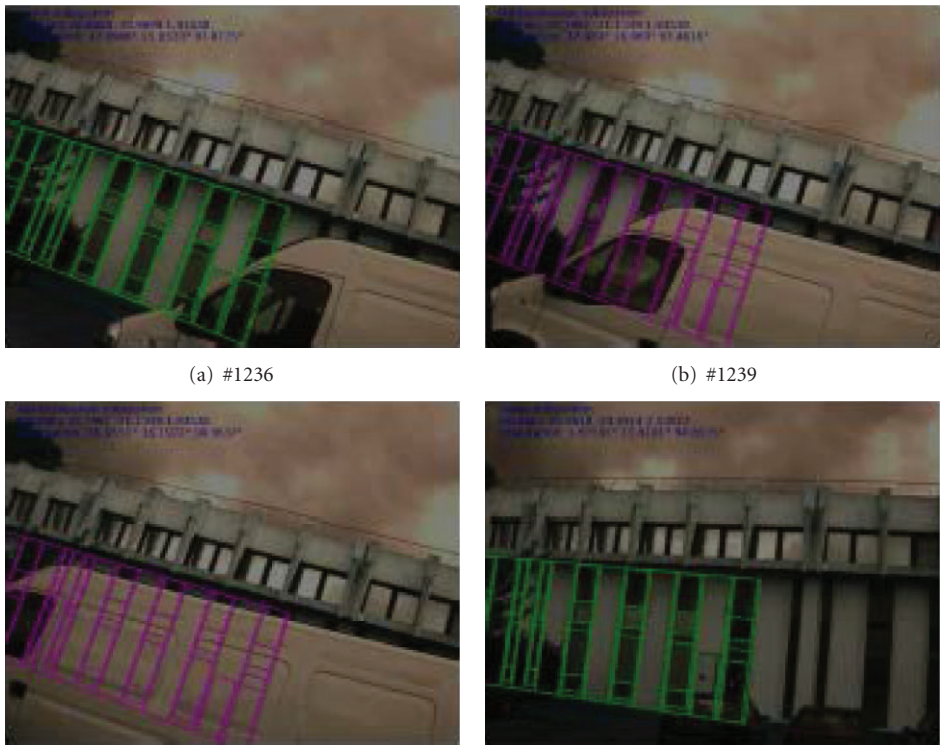
In this paper, we presented an original solution for 3D camera localization using multi-sensors technology. The system combines a camera, a GPS and an inertial sensor; it is designed to work in outdoor environments. Instead to fusion all data, the proposed system is based on an assistance scheme. It is composed of two parts which work



(a) #0686 (b) #1053

(c) #1054 (d) #1055

Figure 15: Registration of the 3D model using the poses obtained with our hybrid system.



(a) #1236 (b) #1239

(c) #1245 (d) #1269

Figure 16: Registration of the 3D model using the auxiliary subsystem: Occlusion case.

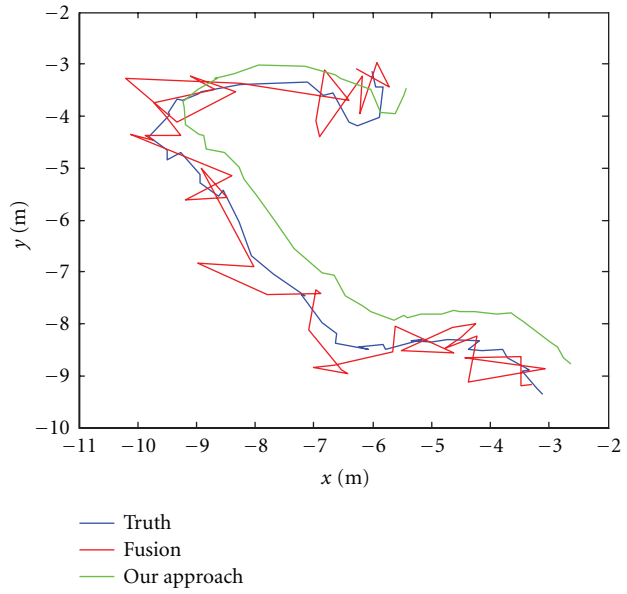


Figure 17: Trajectory in world coordinate system.

in a complementary manner and controlled by a finite state machine allowing continuous 3D localization. The vision subsystem, representing the main part, uses a point-based visual tracking. Once the vision fails, the system switches to Auxiliary subsystem which is composed of the GPS/inertial sensors. The Auxiliary subsystem is less accurate then the vision subsystem, especially the GPS positioning. Hence, a prediction stage is performed to improve the accuracy of the Auxiliary subsystem. The 3D localization provided by the two subsystems is used to learn, on-line, the errors made by the Auxiliary subsystem. The two subsystems interact continuously to each other. The obtained results are quite satisfactory with respect to the purpose of Mobile Augmented Reality systems. They have shown that the proposed system has quite good accuracy compared to other approaches.

The system was tested in outdoor environment and has demonstrated its capacity to adapt itself to the several conditions occurred in such environments. For example, when a total occlusion of the scene model is occurred, the Auxiliary system takes over the 3D localization estimation until the vision becomes operational. However to increase the robustness and the efficiency of the whole system, improvements must be made in several parts. Actually, within the implemented vision-based method, the tracked points must be always visible. So, one challenge is to develop a tracking method which can handle visual occlusions and update automatically the set of tracked points by adding, in real time, new visible points. In addition, other markerless tracking approaches can be combined with the point tracker such as edge-based methods [23] or fiducials based approaches [24, 25] to improve the accuracy of the vision-based pose estimation. Also, the fusion process can be optimized if we consider the motion dynamic of the camera given by the IMU sensor. On the other hand, the experiments have shown that the GPS signal can be obstructed when the user is quite near the buildings. So, when the system switches to the Auxiliary subsystem, the position could not be estimated. This problem can be solved by adding other kinds of positioning sensors which can replace the GPS (RFID, WIFI, etc.). The main idea is to develop a ubiquitous tracking system composed of a network of complementary sensors which can be solicited separately and in real time in terms of the situations occurred in the environments.

References

[1] T. Vieville, F. Romann, B. Hotz et al., "Autonomous navigation of a mobile robot using inertial and visual cues," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 360–367, July 1993.

[2] S. You, U. Neumann, and R. Azuma, "Orientation tracking for outdoor augmented reality registration," *IEEE Computer Graphics and Applications*, vol. 19, no. 6, pp. 36–42, 1999.

[3] M. Ribo, P. Lang, H. Ganster, M. Brandner, C. Stock, and A. Pinz, "Hybrid tracking for outdoor augmented reality applications," *IEEE Computer Graphics and Applications*, vol. 22, no. 6, pp. 54–63, 2002.

[4] J. D. Hol, T. B. Schön, F. Gustafsson, and P. J. Slycke, "Sensor fusion for augmented reality," in *Proceedings of the 9th International Conference on Information Fusion*, pp. 1–6, Florence, Italy, July 2006.

[5] F. Ababsa, "Advanced 3D localization by fusing measurements from GPS, inertial and vision sensors," in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics (SMC '09)*, pp. 871–875, San Antonio, Tex, USA, October 2009.

[6] G. Bleser and D. Stricker, "Advanced tracking through efficient image processing and visual-inertial sensor fusion," in *Proceedings of IEEE International Conference on Virtual Reality (VR '08)*, pp. 137–144, March 2008.

[7] F. Ababsa, J. Y. Didier, M. Mallem, and D. Roussel, "Head motion prediction in augmented reality systems using Monte Carlo particle filters," in *Proceedings of the 13th International Conference on Artificial Reality and Telexixtance (ICAT '03)*, pp. 83–88, Tokyo, Japan, 2003.

[8] F. E. Ababsa and M. Mallem, "Hybrid three-dimensional camera pose estimation using particle filter sensor fusion," *Advanced Robotics*, vol. 21, no. 1-2, pp. 165–181, 2007.

[9] G. Reitmayr and T. W. Drummond, "Initialisation for visual tracking in urban environments," in *Proceedings of the 6th IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR '07)*, Nara, Japan, November 2007.

[10] Z. Hu, U. Keiichi, H. Lu, and F. Lamosa, "Fusion of vision, 3D gyro and GPS for camera dynamic registration," in *Proceedings of the 17th International Conference on Pattern Recognition (ICPR '04)*, pp. 351–354, Washington, DC, USA, August 2004.

[11] M. Aron, G. Simon, and M. O. Berger, "Use of inertial sensors to support video tracking," *Computer Animation and Virtual Worlds*, vol. 18, no. 1, pp. 57–68, 2007.

[12] M. Maidi, F. Ababsa, and M. Mallem, "Vision-inertial tracking system for robust fiducials registration in augmented reality," in *Proceedings of IEEE Symposium Computational Intelligence for Multimedia Signal and Vision Processing (CIMSVP '09)*, pp. 83–90, Nashville, Tenn, USA, April 2009.

[13] D. G. Lowe, "Fitting parameterized three-dimensional models to images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 5, pp. 441–450, 1991.

[14] R. M. Haralick, H. Joo, C. N. Lee, X. Zhuang, V. G. Vaidya, and M. B. Kim, "Pose estimation from corresponding point data," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 19, no. 6, pp. 1426–1446, 1989.

[15] G. Bleser and D. Stricker, "Advanced tracking through efficient image processing and visual-inertial sensor fusion," *Computers and Graphics*, vol. 33, no. 1, pp. 59–72, 2009.

[16] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[17] C. Harris, *Tracking with Rigid Models, Active Vision*, MIT Press, Cambridge, Mass, USA, 1993.

[18] 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*, vol. 24, no. 6, pp. 381–395, 1981.

[19] C. Tomasi and T. Kanade, "Detection and tracking of point features," Carnegie Mellon University Technical report CMU-CS-91-132, 1991.

[20] O. D. Faugeras and G. Toscani, "Camera calibration for 3D computer vision," in *Proceedings of the International Workshop on Industrial Applications of Machine Vision and Machine Intelligence*, pp. 240–247, 1987.

[21] C. Williams, "Prediction with Gaussian processes: from linear regression to linear prediction and beyond," Tech. Rep., Neural Computing Research Group, 1997.

[22] S. DiVerdi and T. Höllerer, "GroundCam: a tracking modality for mobile mixed reality," in *Proceedings of IEEE International Conference on Virtual Reality (VR '07)*, pp. 75–82, March 2007.

[23] F. Ababsa and M. Mallem, "Robust camera pose estimation combining 2D/3D points and lines tracking," in *Proceedings of IEEE International Symposium on Industrial Electronics (ISIE '08)*, pp. 774–779, Cambridge, UK, July 2008.

[24] F. Ababsa and M. Mallem, "A robust circular fiducial detection technique and real-time 3D camera tracking," *Journal of Multimedia*, vol. 3, no. 4, pp. 34–41, 2008.

[25] J. Y. Didier, F. Ababsa, and M. Mallem, "Hybrid camera pose estimation combining square fiducials localisation technique and orthogonal iteration algorithm," *International Journal of Image and Graphics*, vol. 8, no. 1, pp. 169–188, 2008.