Signal Processing 90 (2010) 3294–3300

Contents lists available at ScienceDirect

Signal Processing

journal homepage: www.elsevier.com/locate/sigpro

Fast communication

Low-cost GPS sensor improvement using stereovision fusion

David Schleicher *, Luis M. Bergasa, Manuel Ocaña, Rafael Barea, Elena López

Department of Electronics, University of Alcala, Alcalá de Henares, 28805 Madrid, Spain

ARTICLE INFO

Article history:
Received 3 March 2010
Received in revised form
28 April 2010
Accepted 12 May 2010
Available online 19 May 2010

Keywords:
GPS sensor
Intelligent vehicles
Computer vision
Sensor fusion

ABSTRACT

This paper presents a new real-time hierarchical (topological/metric) localization system applied to the robust self-location of a vehicle in large-scale urban environments. Our proposal improves the current vehicle navigation systems based only on GPS sensor. It is exclusively based on the information provided by both, a low-cost wide-angle stereo camera and a low-cost GPS. A low level metric process obtains a 3D sequential mapping of natural landmarks and the vehicle location/orientation. GPS measurements are integrated within this low level, improving vehicle positioning. A higher topological processing level, based on fingerprints and the multi level relaxation (MLR) algorithm, has been added to reduce the global error keeping real-time constraints. Some experimental tests, using a real car navigation system on urban environments with loop closures, have been carried out. Main results and conclusions are presented.

1. Introduction

Low-cost vehicle navigation systems based on GPS are installed in most of the vehicles nowadays. One of the main problems of these systems is its performance degradation on urban environments. High buildings, tunnels, etc, can reduce vehicles location accuracy and they can even cause data loss, making navigation unavailable. One of the main goals of this work is to improve vehicle navigation, mainly in very populated urban areas where the GPS information is not reliable. On-board navigation systems solve this problem by coupling dead-reckoning and GPS positions. Dead-reckoning is calculated through an on-board low-cost inertial measurement unit (IMU), which estimates vehicle's movements, and the covered distance estimation, available through ABS wheels speed sensors. Some basic references about GPS and inertial navigation can be found in [1,2]. Despite many recent improvements in the characterization of acceleration and rotation rate, measurement errors due to thermal stability of micro-electro-mechanical-systems (MEMS) components, cause drifts in pure integration cycle even with the aid of odometry. Fiber-optic gyrometers offer more accuracy than MEMS, but their costs do not comply with automotive cost requirements [3]. In the last years, an alternative to GPS for absolute localization has been to use active beacons with known locations. Recently many researchers have been working on adapting localization and mapping strategies from the robotic to the vehicle localization problem using different sensors fusion. In [3] a vehicle is able to navigate and self-locate using a GPS, an inertial navigation system (INS) and odometry. A multi-model interactive multi-model EKF (IMM-EKF) is applied for fusion information. A highly accurate estimation is achieved, but with an expensive solution. In [4] a self-localization system for outdoor environments is presented. The vehicle is equipped with a stereo camera, an IMU, odometry as well as a standard GPS. The vehicle is able to self-locate with relatively low error within medium size environments (around 100 m). However, the fact that the system does not use any specific management method for large environments limits the use on this kind of environments, even more

* Corresponding author.
E-mail address: dsg68818@gmail.com (D. Schleicher).

0165-1684/$ - see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.sigpro.2010.05.017

in the case of IMU, GPS or odometry unavailability. Cameras have become much more inexpensive than lasers or radar, and also provide texture rich information about scene elements at practically any distance from the camera. Our proposal consists on obtaining vehicle dead-reckoning using visual information from a stereo camera, instead of an IMU, due to our goal to develop a low-cost standard nomadic system independent of the manufacturer
protocols confidentiality. Moreover, as difference of IMU systems, our proposal generates a map and it is able to detect loop closings using visual appearance information. Then, scenarios with recurrent trajectories, as buses routes, are optimal for this system. In this way, accumulated drifts, typical of odometry sensors, are removed from time to time even with GPS unavailability. So, to improve low-cost vehicle navigation systems performance, our proposal is able to provide a vehicle navigator with pose data by fusing information from both a low-cost stereo camera and a low-cost GPS. To deal with the large-scale environments problem, the system is divided into two hierarchical processing levels.

2. Implementation

Hardware architecture of our system is depicted in Fig. 1. From a processing point of view, our approach defines local metric sub-maps independent among them. They are composed by several visual landmarks and managed by an EKF, on what we call the low processing level. Over these local sub-maps we define a higher topologic map level, called high processing level, which relates the local sub-maps keeping the global map consistency (see Fig. 2). On one hand, low processing level is based on a visual system using a low-cost stereo wide-angle camera mounted on the windshield area of a vehicle and looking forward of the vehicle. An EKF filter is used to obtain the visual estimation. On the other hand, GPS measurements are taken at the same time, contributing to reduce the accumulated drift of the system when no loops are closed.

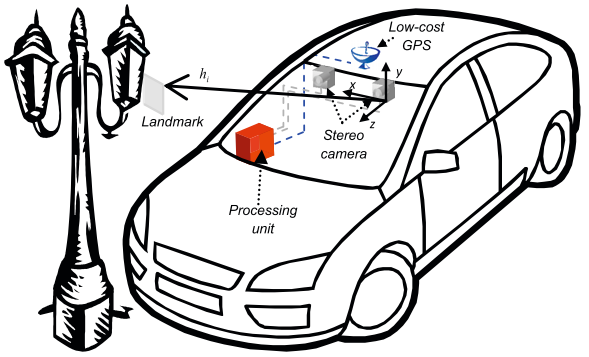

Fig. 1. Hardware architecture mounted on a commercial car. Stereovision system and low-cost GPS as well as the processing unit are shown. The *measurement prediction* vector h_i as well as the vehicle reference frame are also depicted.

2.1. Low processing level

This level implements algorithms and tasks needed to locate and map the vehicle on its local sub-map using visual information. It is based on the monocular approach by Davison [5] and its adaptation to stereo developed by the authors [6]. For clarity reasons the sub-map notation is omitted, so it is assumed a unique sub-map for the low processing level implementation. In Fig. 2 (left) we show the main tasks carried out on the low processing level. It essentially implements an EKF where a prediction on the vehicle+landmarks locations is updated with the visual observation. This estimation is again improved with the low-cost GPS measurement estimation.

2.1.1. Extended Kalman filter application

In order to apply an EKF, a state vector X and its covariance matrix P need to be defined. The purpose of the algorithm is to continuously estimate the position and orientation of the vehicle, via the linearization of the *next state function*, $f(X)$, at each time step. Vehicle coordinate system has been set in the camera frame one. Due to the *motion model* used for the vehicle movement, linear and angular speeds are added to the vehicle state vector: $X_v=(X_{vh},q_{vh},v_{vh},\omega)^T$. In this equation, $X_{vh}=(x_{vh},y_{vh},z_{vh})^T$ is the 3D position of the camera relative to the global frame, $q_{vh}=(q_0,q_x,q_y,q_z)^T$ is the orientation quaternion, v_{vh} is the linear speed and ω is the angular speed. On the other hand, as the whole sub-map has to be included into the filter, all features global positions, Y_i, are added to the state vector: $X=(X_v,Y_1,Y_2\cdots)^T$.

2.1.2. Motion model

To build a motion model for a camera mounted on a mobile vehicle using only visual information, a practical solution is to apply the so-called *impulse model*. This assumes constant speed (both linear and angular) during each time step and random speed changes between steps in the three directions. Some restrictions have been applied to adapt the 6DOF generic model to the vehicle's dynamics. According to this model, to predict the next state of the camera, the function $f_v=(X_{vh}+v_{vh}\,\Delta t,q_{vh}\times q[\omega\,\Delta t]v_{vh},\omega)^T$ is applied. The term $q[\omega\,\Delta t]$ represents the transformation of a 3 components vector into a *quaternion*. Assuming that the map does not change during the whole process, the absolute feature positions Y_i should be the same from one step to the next one. This model is subtly effective and gives the whole system important robustness even when visual measurements are sparse.

2.1.3. Measurement model

Hereafter we present a brief description of our proposal, for a deeper explanation we remit the reader to [6]. Visual measurements are obtained from the "visible" features positions. In our system we define each individual *measurement prediction* vector $h_i=(h_{ix},h_{iy},h_{iz})^T$ as the corresponding 3D feature position relative to the camera frame (see Fig. 1). To choose the features to measure, some selection criteria have to be defined. These criteria will be based on the feature "visibility"; that is,

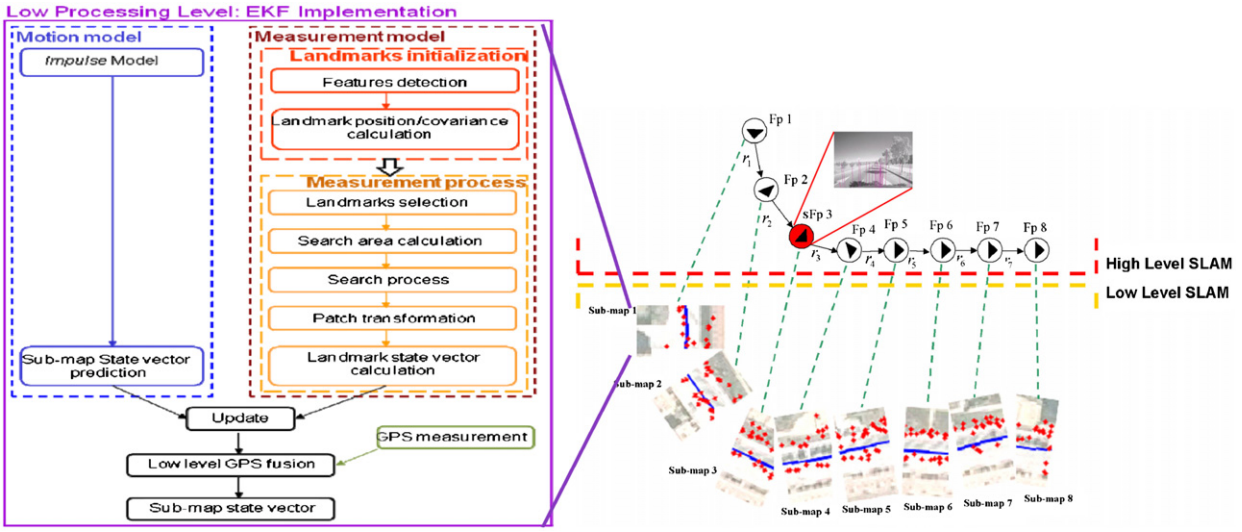

Fig. 2. (*Left*) Low processing level tasks carried out within each sub-map. (*Right*) General architecture of our two hierarchical processing levels. Each sub-map has an associated fingerprint.

whether its appearance is close enough to the original one (when the feature was initialized). This is based on the relative distance and point of view angle respect to the one at the feature initialization phase. The first step is to predict the measurement vector h_i. To look for the actual measurement vector z_i, we have to define a search area on the projection images. This area will be around the projection points of the predicted measurement h_i on both *left* and *right* images: U_L: (u_L,v_L), U_R: (u_R,v_R). To obtain the image projection coordinates a simple camera pin-hole model is applied. Radial and tangential distortion models, calculated for the cameras in a calibration setup, are then carried out. To obtain z_i we solve the inverse geometry problem applying the distortion models as well. Regarding the search areas, they will be calculated based on the uncertainty of the feature 3D position, which is called *innovation covariance* S_i (see [5]). As we have two different image projections, S_i needs to be transformed into the projection covariance P_{UL} and P_{UR} using Eq. (1)

$$P_{U_L}=\frac{\partial U_L}{\partial h_i}S_i\left(\frac{\partial U_L}{\partial h_i}\right)^T;\quad P_{U_R}=\frac{\partial U_R}{\partial h_i}S_i\left(\frac{\partial U_R}{\partial h_i}\right)^T \tag{1}$$

These two covariances define both elliptical search regions, which are obtained taking a certain number of standard deviations (usually 3) from the 3D Gaussians. Once the areas where the current projected feature should lie are defined, we can look for them. At the initialization phase, the left and right images representing the feature *patches* are stored. Then, to look for a feature patch, we perform normalized *sum-of-squared-difference correlations* across the whole search region. In order to improve long-term tracking, a 2D patch warping is done considering the normal vector information of each patch.

2.1.4. Feature initialization

The selected criteria to initialize new landmarks are to maintain always at least 5 visible features and 4 successfully measured features. Then, when a new feature initialization needs to take place, its corresponding patch will be searched within a rectangular area randomly located on the left camera image. To obtain the right image feature correspondence we search over the *epipolar line*, restricted to a certain segment around the estimated right projection coordinates.

2.1.5. GPS fusion

To reduce the error in the vehicle pose estimations, caused by the accumulated drift, visual estimation is fused with the GPS pose information by using a statistical approach [7], as shown in (2). In this equation X_{Pvh},P_{Pvh} stands for the vehicle 2D position and covariance calculated from the visual information, and X_{GPS},P_{GPS} for the vehicle 2D position and covariance obtained from the GPS sensor

$$X^{fusion}=X_{Pvh}+P^G_{Pvh}(P^G_{Pvh}+P_{GPS})^{-1}(X_{GPS}-X_{Pvh}) \tag{2}$$

In the same way, the fused vehicle estimated covariance is calculated by mean of Eq. (3)

$$P^{fusion}=P^G_{Pvh}-P^G_{Pvh}(P^G_{Pvh}+P_{GPS})^{-1}P^G_{Pvh} \tag{3}$$

To obtain the final pose, including orientation, interpolation on the two last GPS updates is carried out.

2.2. High processing level

To reduce global error when GPS is not available, a high topological processing level, based on fingerprints, is added. This level defines a topological map, where each node (fingerprint) $FP=\{fp_l|l\in 0\cdots L\}$ is associated to a segment of the path covered by the vehicle. These segments are called sub-maps. The fingerprints store the vehicle pose at the moment of the sub-map creation and they define its local reference frame. In Fig. 3 (left) we show the overall diagram for this level. The sub-map generation is performed periodically so, after a certain covered section of the path, a new sub-map is created and

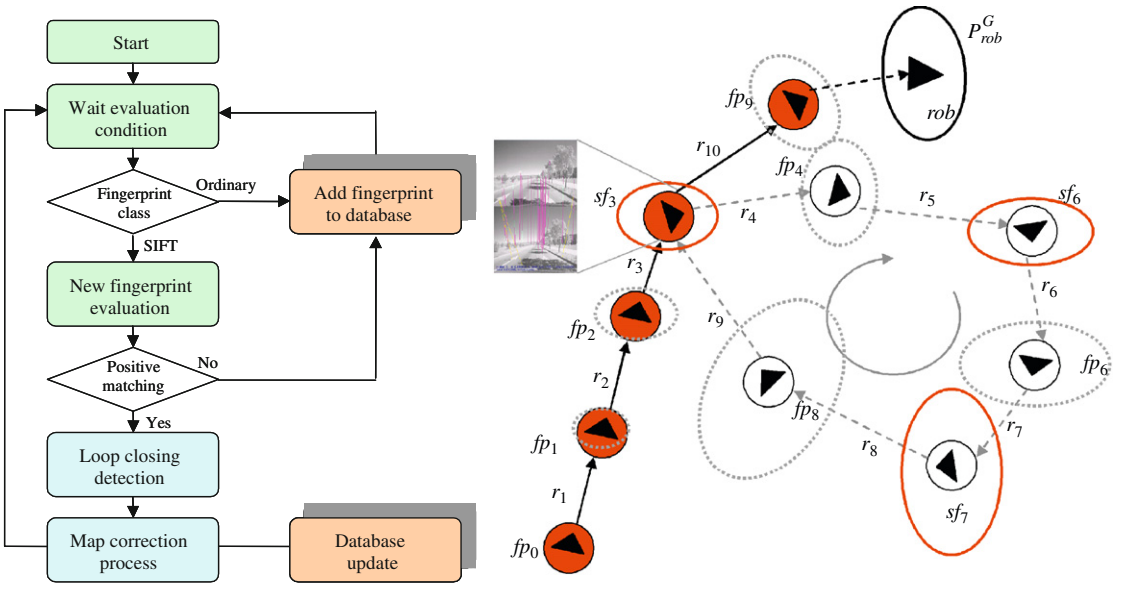

Fig. 3. (*Left*) High processing level diagram. Each ordinary fingerprint is added periodically to the database. In case a SIFT fingerprint condition is detected, it is created and compared with previous SIFT fingerprints. In case of positive matching a loop closing + map correction process is generated. (*Right*) Representation of the high level topological map. Vehicle global uncertainties P^G_{rob} are shown, increasing along the vehicle path at each of the fingerprints poses. Solid red lines represent vehicle global uncertainties at SIFT fingerprints places. Numbers represent each fingerprint. Graph also shows an example of shorter path selection for global uncertainty calculation after a loop closing situation.

a fingerprint is associated to it. If the vehicle is traveling between two fingerprints, an edge is inserted to connect these two vertices, which represents a link between two poses. Meanwhile, the edges store transformation matrices $X^{fp_l}_{vh}$ and uncertainties $P^{fp_l}_{vh}$ to describe the relationship between connected fingerprints, as Fig. 3 (right) depicts.

To carry out the loop closing detection when the GPS is lost, an additional type of fingerprint called scale invariant feature transform (SIFT) fingerprint $SF=\{sf_q \in FP | q \in 0\cdots Q, Q<L\}$ is defined. This is based on a variable-size set of SIFT features $YF^q=\{Yf^q_m | m \in 0\cdots M\}$ and it is taken when a significant vehicle turn is detected. This adds to the vehicle pose some visual information to identify the place where it was taken. Matching between the previously captured SIFT fingerprints, within an uncertainty area, and the current one is carried out to detect pre-visited zones. We calculate this uncertainty area P^G_{vh} by expressing the current local uncertainty $P^{fp_l}_{vh}$ in the global reference frame, where X^0_{vh} is the current global vehicle position

$$P^G_{vh}=\frac{\partial X^0_{vh}}{\partial X^{fp_i}_{vh}} P^{fp_l}_{vh} \left(\frac{\partial X^0_{vh}}{\partial X^{fp_i}_{vh}}\right)^T \tag{4}$$

In case of positive matching, a loop closing is detected and the topological map is corrected by using the MLR algorithm [8] over the whole set of fingerprints. The MLR determines the maximum likelihood estimate of all fingerprint poses. The MLR algorithm manages only 2D information, therefore we need to obtain the 2D relative fingerprint pose $X^{fp_{l-1}}_{2fp_l}$ and covariance $P^{fp_{l-1}}_{2fp_l}$ from the corresponding 3D relative fingerprint pose $X^{fp_{l-1}}_{fp_l}$ and covariance $P^{fp_{l-1}}_{fp_l}$. First, the 2D pose is defined as: $X^{fp_{l-1}}_{2fp_l}=\left(x^{fp_{l-1}}_{2fp_l} \quad y^{fp_{l-1}}_{2fp_l} \quad \theta^{fp_{l-1}}_{2fp_l}\right)^T$, i.e., the 2 planar coordinates and the orientation angle. Therefore we can relate both 2D and 3D poses according to Eq. (5).

$$X^{fp_{l-1}}_{2fp_l}=\left(x^{fp_{l-1}}_{fp_l} \quad z^{fp_{l-1}}_{fp_l} \quad 2\arccos\left(q^{fp_{l-1}}_{0\,fp_l}\right)\right)^T \tag{5}$$

where $x^{fp_{l-1}}_{fp_l}$, $z^{fp_{l-1}}_{fp_l}$ and $q^{fp_{l-1}}_{0fp_l}$ are coordinates of $X^{fp_{l-1}}_{fp_l}$. Also, we compute the 2D covariance by using the corresponding jacobians. On the other hand, if the GPS signal was missed, once it is recovered again, the global vehicle pose is updated and the global map is corrected. Its effect is similar to a loop closing detection, as shown in Fig. 4 (right). Tasks carried out at the high processing level are slower than the ones at the low processing level. Therefore they are parallelized and the final processing time is kept below the real-time constraint, defined at 33 ms per step for this work. With our proposal a global map optimization is carried out even when the GPS signal is lost for a long term. Finally, the obtained vehicle pose data is formatted using the NMEA protocol and continuously sent to the car navigator.

3. Results

The system has been tested on several urban paths, usually covered by a public bus line. Fig. 5 shows one of these paths. This is 1.41 km long and presents 3 areas where the GPS is lost. Especially in these areas, our proposal improves the car navigation performance based only on GPS. Fig. 6 depicts the error respect to the ground truth on X and Z axes, using the standalone GPS and our

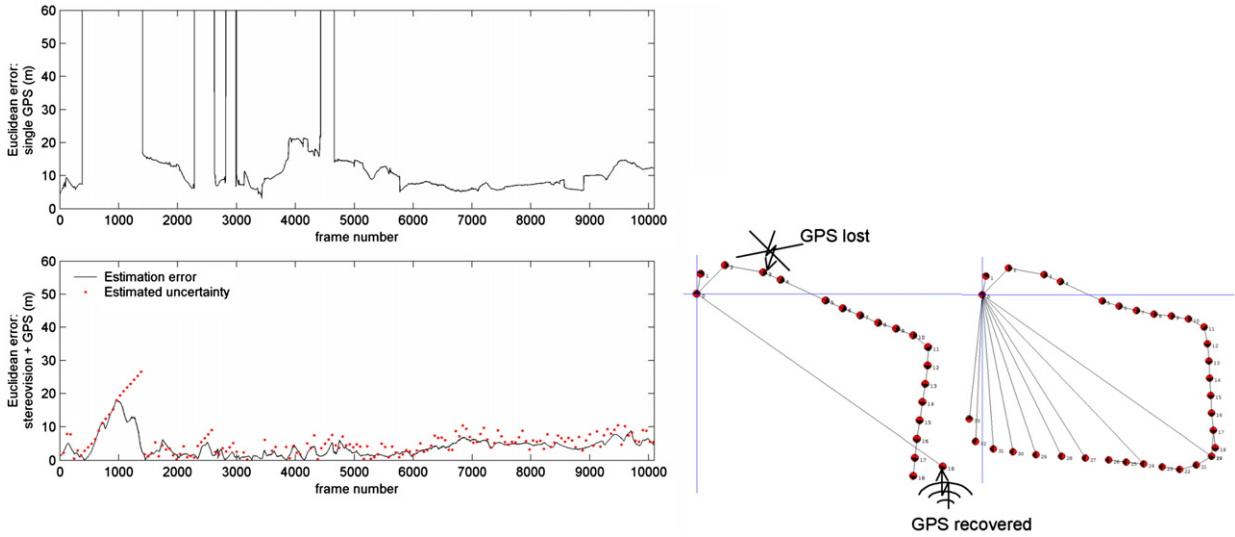

Fig. 4. (*Left*) Euclidean distance error ($\varepsilon = \sqrt{X^2+Z^2}$) using standard single GPS (up) and our combined system (down). Global covariances uncertainties for each node are shown as well. (*Right*) MLR diagram before (left) and after (right) GPS recovering. When GPS misses, the vehicle pose is estimated by using vision only. Some drift is appreciated due to the *relative measurement* nature of visual estimation combined with unpredictable errors, as a consequence of partial occlusions from great vehicles, for example. Error estimation in these cases is minimized by the use of wide-angle lenses, which give wide fields of view.

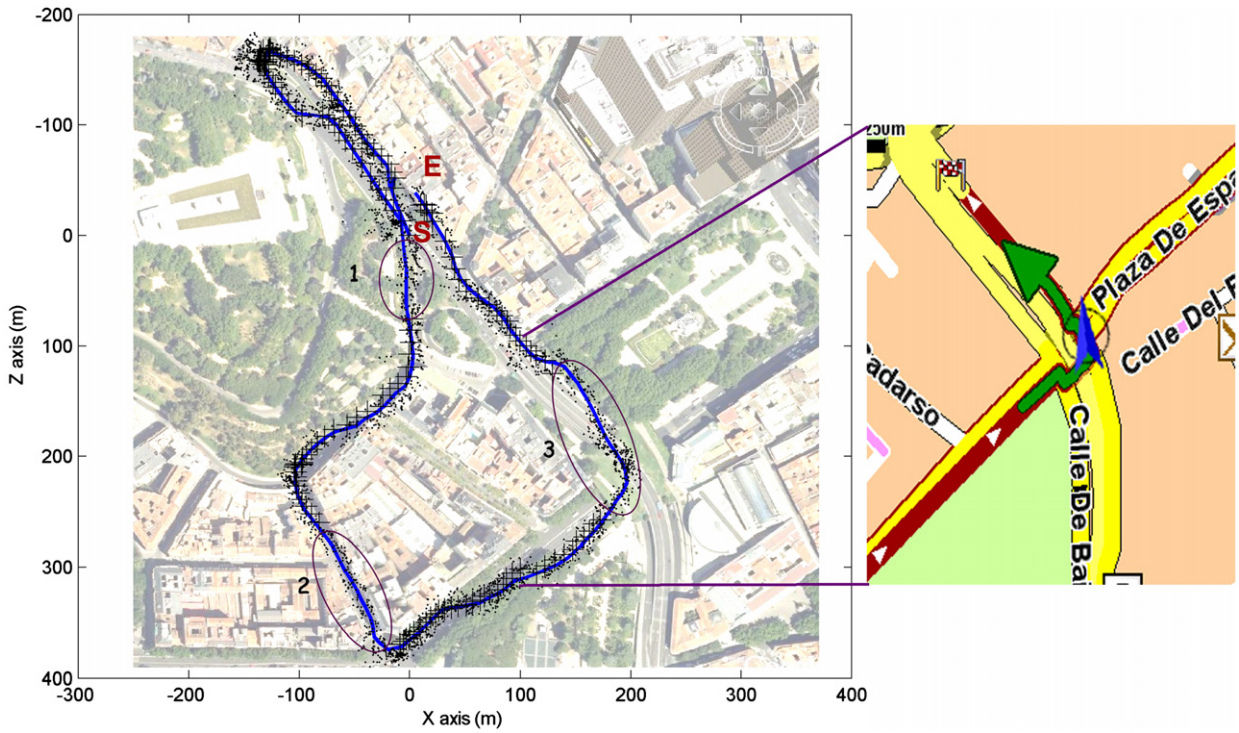

Fig. 5. (Left) Path estimation using our system. Crosses indicate GPS measurements and black dots represent visual landmarks. Numbered ellipses indicate the areas where GPS signal was lost. (Right) Vehicle turn to the left within a tunnel correctly interpreted by the navigator in the absence of GPS information. Start point is marked as *S* and end point as *E*.

combined system. Ground truth is obtained using an RTK-GPS Maxor GGDT, with an estimated accuracy of 2 cm.

Focusing on the third GPS loss, the vehicle was in an underground crossroad. At this time the vehicle turns to the left. As the system still has visual information available, the navigator is able to realize about the vehicle turn and perform the correct path planning. Results of another path are presented. In this case, it was 3.17 km long and contained 5 loops inside, taking 8520 low level landmarks and 281 nodes. Most landmarks were located

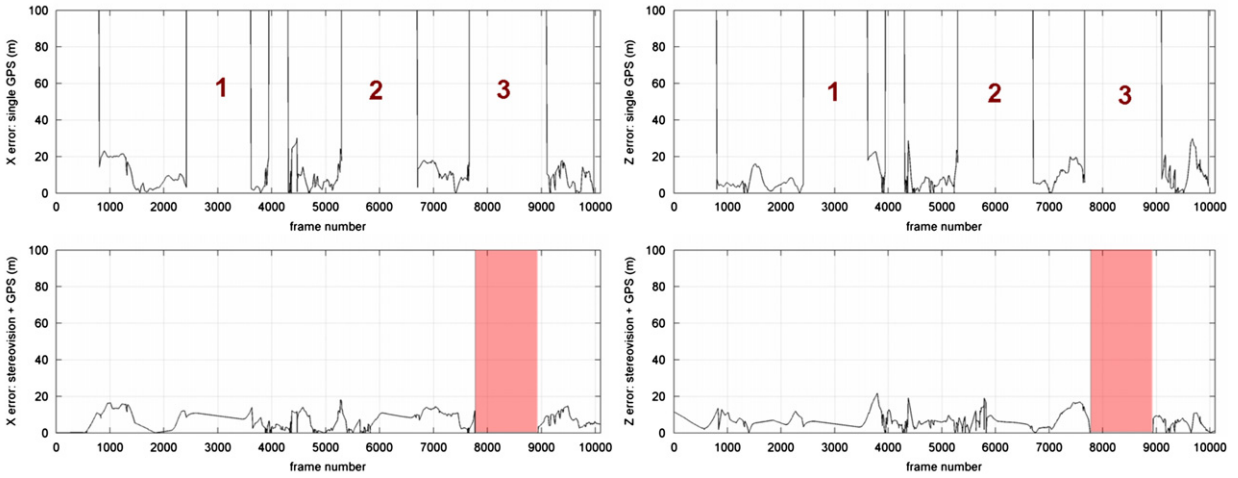

Fig. 6. Path estimation error on *X*-axis (left) and *Z*-axis (right), using a standalone GPS (up) and our system (down). Numbers indicate GPS loss sections shown in Fig. 3. The red marked area indicates the third GPS loss, where ground truth was also unavailable.

Table 1
Processing times.

Low processing level computation times		High processing level computation times	
Number of features/frame	5	Number of features	8520
		Number of fingerprints	281
Filter step	Time		Time
Measurements	3 ms	Fingerprint matches	3 s
Filter update	5 ms	Loop closing	1 s
Feature initializations	7 ms		
GPS processing (1 s sampling period)	4 ms		

on high buildings areas, where GPS is less reliable, while GPS signal is stronger in open-spaced areas. In these last areas better location estimation is provided due to GPS data correct visual data deviations from time to time. As it can be seen, both sensors are complementary, providing good estimations for different situations. The Euclidean error relative to the ground truth of both the standard GPS and our combined implementation is depicted in Fig. 4. Some videos of the approach working for different paths can be found on our website[1] . We obtain an average error around 4 m and a reasonable low error at the moments of total GPS loose. This error is compared to the global uncertainty covariances for each node using the Euclidean formula applied to the *X* and *Z* components as well, showing consistent error estimates. As expected, uncertainty monotonically grows on GPS unavailable sections due to the relative measurements provided by the visual sensor. More than 20 km were tested with this method and obtained results were similar to that shown here. All test cases show a global processing time below the real-time constraint (33 ms/frame). Table 1 shows the processing times per each task in a detailed way.

It shall be taken into account that tasks within the high processing level are carried out only upon certain events such as: fingerprint evaluation, loop closing detection or after loss of GPS signal. Then, they can be processed in parallel with low processing level tasks and, as a consequence, under real-time constraint. Robustness of our proposal has been shown to be enough for an application of vehicles localization in urban environments which include trajectories with loop closures. It was tested at different times along the day with different light and traffic conditions. Positioning errors were low at day time, regardless of traffic conditions. At night time, results get worse due to low lighting conditions. Several limitations were identified at the time of practical implementation. Vehicle's positioning error increases in the following cases: large tunnels with poor textures or low illumination, long unavailability of GPS without loops in the trajectory, loop closing from very different trajectories (points of view) and long routes with poor lighting conditions.

4. Conclusion

We have presented a robust localization system based on the fusion of visual information and GPS data. Our

[1] http://www.robesafe.com/tecnologias/index_en.php#robotics.

proposal improves the behavior of a standard nomadic vehicle navigator system. Results showed a reduced estimation error with respect to the use of a standalone GPS sensor, mainly in GPS loss areas but also in the whole sequence. Improvements on the navigator behavior are shown as well on a practical implementation tested on urban environments, which include trajectories with loop closures similar to the buses routes. Some limitations have been identified for our system. As future work we plan to overcome some of these limitations.

Acknowledgement

This work has been funded by Grant TRA2008-03600/AUT (DRIVER-ALERT project) from the Spanish Ministry of Science and Innovation (MICINN), and Grant S2009/DPI-1559 (Robocity2030-II project) from the Science Department of the Community of Madrid.

References

[1] J.A. Farrell, M. Barth, in: The Global Positioning System and Inertial Navigation, Mc Graw Hill, New York, 1999.
[2] M.S. Grewal, L.R. Weill, P. Andrews, in: Global Positioning Systems, Inertial Navigation and Integration, Wiley Interscience, New York, 2001.
[3] R. Toledo-Moreo, M. Zamora-Izquierdo, B. Ubeda-Miarro, A. Gomez-Skarmeta, High-integrity IMM-EKF-based road vehicle navigation with low-cost gps/sbas/ins,, IEEE Transactions on ITS 8 (3) (2007) 491–511.
[4] M. Agrawal, K. Konolige, Real-time localization in outdoor environments using stereo vision and inexpensive gps, ICPR '06: Proceedings of the 18th International Conference on Pattern Recognition, 2006, pp. 1063–1068.
[5] A.J. Davison, Real-time simultaneous localisation and mapping with a single camera, ICCV03 (2003) 1403–1410.
[6] D. Schleicher, L.M. Bergasa, M. Ocaña, R. Barea, E. Lopez, Real-time hierarchical GPS aided visual SLAM on urban environments, ICRA (2009) 4381–4386.
[7] A.W. Stroupe, M.C. Martin, T. Balch, Distributed sensor fusion for object position estimation by multi-robot systems, ICRA2001 2 (2001) 1092–1098.
[8] U. Frese, P. Larsson, T. Duckett, A multilevel relaxation algorithm for simultaneous localization and mapping, IEEE Transactions on Robotics 21 (2) (2005) 196–207.