Optimal Metric SLAM for Autonomous Navigation Assistance

P.F. Alcantarilla, L.M. Bergasa, I. Parra, D. Schleicher
Department of Electronics. University of Alcalá
Alcalá de Henares (Madrid), Spain
pablo.alcantarilla, bergasa, parra@depeca.uah.es, dsg68818@telefonica.net

***Abstract*— In this paper we present a 6DOF metric SLAM system for outdoor enviroments using a stereo camera, mounted next to the rear view mirror, as the only sensor. By means of SLAM the vehicle motion trajectory and a sparse map of natural landmarks are both estimated at the same time. The system combines both bearing and depth information using two different types of feature parametrization: inverse depth and 3D. Through this approach near and far features can be mapped, providing orientation and depth information respectively. Natural landmarks are extracted from the image and are stored as 3D or inverse depth points, depending on a depth threshold. At the moment each landmark is initialized, the normal of the patch surface is computed using the information of the stereo pair. In order to improve long-term tracking a 2D warping is done considering the normal vector information of each patch. This Visual SLAM system is focused on the localization of a vehicle in outdoor urban environments and can be fused with other cheap sensors such as GPS, so as to produce accurate estimations of vehicle's localization in a road. Some experimental results under outdoor environments and conclusions are presented.**

I. Introduction

Real-time Simultaneous Localization and Mapping has an important key role in robotics. In recent times, SLAM has captured the attention of computer vision researchers and the interest of using cameras as sensors has grown considerably due to mainly three reasons. Cameras are cheaper than commonly used scan-lasers, they provide rich visual information about scene elements and are easy to adapt for wearable systems. According to that, the range of SLAM based applications has spread to non typical robotic environments such as augmented reality [1], non-invasive surgery [2] and vehicle localization [3].

In this work a 6DOF Stereo SLAM system is proposed in order to develop a robust localization system, using only a cheap stereo camera mounted next to the rear view mirror, able to complement a standard GPS sensor for autonomous vehicle navigation where GPS signal does not exist or it is not reliable (tunnels, urban areas...). At the same time, a sparse map of high quality features is computed. This optimized map contributes to a better localization estimate and prevents the system from drifting in situations where the vehicle visits some areas that were previously visited, i.e. loop closing situations. The main advantages of using a stereo system instead of a monocular one were described in [4].

The traditional approach in in the literature for solving the SLAM problem, is using an extended Kalman Filter (EKF) with the vehicle pose and static landmarks as the evolving filter state. This EKF approach has some drawbacks as it is explained in [5]. The main drawback of the EKF implementation is that the computational requirement for the filter update increases quadratically in large-scale maps as a function of the landmarks introduced into the filter $O(n^2)$. A typical solution to cope with this problem is submapping, where the global map is obtained fusing the information from local submaps [3], [6].

Our system follows a Davison's SLAM approach [7]. That is, a few high quality features are tracked and used to compute the position of the camera creating a sparse map of high quality textured landmarks using an Extended Kalman Filter (EKF). Paz et al. proposed in [8] a 6DOF Stereo EKF-SLAM system with stereo in hand for large indoor and outdoor environments. The inverse depth parametrization proposed by Civera et al. [9] for the MonoSLAM approach is adapted to the StereoSLAM version so as to provide distance and orientation information. Point features are extracted from the images and are classified as 3D features if the disparity is enough, or stored as inverse depth features otherwise. Their Visual SLAM algorithm generates conditionally independent local maps and finally, the full map is obtained using the Conditionally Independent Divide and Conquer algorithm, which allows constant time operation most of the time [6]. Although results are good considering large maps in indoor/outdoor environments, the range of camera movements is limited, since no patch adaptation is done and only 2D image templates correlations are carried out in the matching process. By means of an empirical analysis, they suggest choosing a threshold of depth 5 m, for switching between inverse depth and 3D features. Besides, our sequences are more suitable to show the benefits of an inverse depth parametrization for far features.

The accuracy of the stereo sensor is limited up to a certain depth, depending mainly on the baseline of the sensor. In typical outdoor road vehicles sequences, is common to have very far landmarks. If we try to measure the 3D position of a far feature, which is located beyond the limits of our sensor, the uncertainty in the measurement will be very high. On the contrary we can

reduce the uncertainty of far features if we just measure the orientation of the feature.

The two key contributions of our work, are the use of inverse depth and 3D features for providing both depth and angular information, and a 2D homography warping method considering information from both cameras of the stereo pair. This paper is organized as follows: the general structure of the system is explained in section II. In section III we deal with the problem of how to switch between inverse depth or 3D parametrization. In The 2D homography warping for patch adaptation is described in section IV. Finally, some experimental results are presented in section V. Main conclusions and future works are discussed in section VI.

II. System Structure

Our system is based on a stereo camera mounted on a mobile vehicle close to the rear view mirror. Fig. 1 depicts the common type of sequences in outdoor road vehicle navigation. As it can be observed, some features are very far with respect to the camera, whereas we can have some features close to the camera. Both far and close features are displayed in orange (weak) and red (dark) respectively in Fig. 1.

Fig. 1. Typical outdoor road navigation sequences

The global state vector X incorporates the information for the left camera and for the features. The camera state is composed of its 3D position using cartesian coordinates, the camera orientation in terms of a quaternion, and the linear and angular speed, which are necessary for the impulse motion model used for modelling the camera movement. The motion model that is assumed is a constant velocity and constant angular velocity model explained in [7].

$$\mathbf{X_v}_{\,[\mathbf{13},\mathbf{1}]} = (X_{cam},\ q_{cam},\ v_{cam},\ \omega_{cam})^t \qquad (1)$$

Two types of feature parametrization are used providing orientation and depth information respectively. Depending on the depth of the feature as described in section III, features are initialized as inverse depth or 3D and are incorporated to the EKF SLAM algorithm.

$$\mathbf{X} = (X_v, Y_{1\ 3D} \cdots Y_{n\ 3D}, Y_{1\ INV} \cdots Y_{m\ INV})^t \qquad (2)$$

Interesting points are extracted from the image using the Harris corner detector [10] and a subsequent subpixel refinement. When the camera moves, these features are tracked over the time to update the filter. In order to track a feature, image position is predicted in both cameras. Then, the feature appearance is transformed using a 2D homography according to section IV, and a correlation search is performed inside a search area of high probability which is defined by the uncertainties of the feature and the camera. ZMCC (Zero Mean Cross Correlation) is used since its robustness against lighting changes. An intelligent feature management is implemented, so low-quality features are deleted from the state vector.

Due to the use of a wide-angle lens, it is necessary to use a distortion model correcting distorted images. Unlike other SLAM systems [4], [7] radial and tangential distortion are corrected using LUT (Look up tables), so images are corrected previous to processing. Two main advantages are obtained from using LUTs: firstly, this method is faster than working with the distorted images and then correcting the distorted projection coordinates, and secondly, the matching process is less critical if undistorted images are used.

A. 3D Features

For 3D features, the feature's state vector encodes the information about the 3D position of the feature in the global map reference system.

$$\mathbf{Y_{3D}}_{\,[\mathbf{3},\mathbf{1}]} = (x,\ y,\ z)^t \qquad (3)$$

B. Inverse depth Features

For inverse depth features, the feature's state vector encodes the information of the 3D optical center pose from which the feature was first seen X_{ori}, the orientation of the ray passing through the image point (angles of azimuth θ and elevation ϕ) and the inverse of its depth, ρ. Fig. 2 depicts the inverse depth point coding:

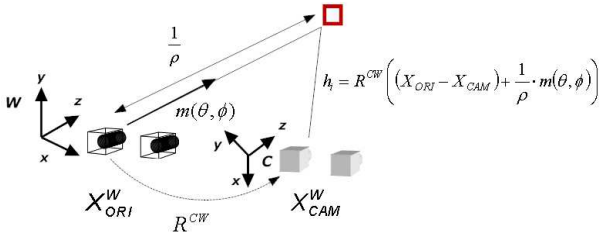


Fig. 2. Inverse depth point coding

$$\mathbf{Y_{INV}}_{\,[\mathbf{6},\mathbf{1}]} = (X_{ori},\ \theta,\ \phi,\ \rho)^t \qquad (4)$$

In Fig. 2, $m(\theta, \phi)$ is the unitary ray directional vector from the camera to the feature. This unitary vector is defined according to eq. 5:

$$\mathbf{m}(\theta,\phi)_{\ [\mathbf{3,1}]} = (\sin\phi\cos\theta, -\cos\phi, \sin\phi\sin\theta)^t \quad (5)$$

The angles of azimuth and elevation are defined as follows:

$$\theta = tan^{-1}\left(\frac{z}{x}\right) \quad (6)$$

$$\phi = tan^{-1}\left(\frac{\sqrt{x^2+z^2}}{y}\right) \quad (7)$$

III. Switching between Inverse Depth and 3D Features

Harris corners are extracted from the images and are classified as 3D features or stored as inverse depth features, depending on a depth threshold. This depth threshold is empirically set to 30 m. The value of this threshold is chosen as a compromise between non-linearity measurements, features uncertainty and the overhead introduced by the inverse depth parametrization. After some experimental tests we found the value of 30 m as a good threshold for our application.

Once the features are predicted in the EKF prediction step, it is necessary to determine if the original parametrization of the features has to be changed (i.e. if an inverse depth feature is now below the depth threshold and should adapt a 3D parametrization or viceversa). Besides, a constraint is imposed: the feature has to remain at least m frames (typically 15 frames) in its new parametrization state before the switching. This is done in order to avoid unnecessary switchings in case that the depth estimate is above and below the threshold in consecutive frames.

When an inverse depth feature is switched to a 3D parametrization, it is necessary to adapt the feature's state and the covariances implied in the filtering process by means of equations 8 for the feature's state and 8, 10 for the covariances. In the same way we can switch between 3D features to inverse depth, although this is not a common case in autonomous navigation.

$$\mathbf{Y_{3D}}_{\ [\mathbf{3,1}]} = X_{ORI} + \frac{1}{\rho}\cdot m(\theta,\phi) \quad (8)$$

$$\mathbf{P_{YY_{3D}}}_{\ [\mathbf{3,3}]} = \left(\frac{\partial Y_{3D}}{\partial Y_{INV}}\right)\cdot P_{YY_{INV}}\cdot\left(\frac{\partial Y_{3D}}{\partial Y_{INV}}\right)^t \quad (9)$$

$$\mathbf{P_{XY_{3D}}}_{\ [\mathbf{13,3}]} = P_{XY_{INV}}\cdot\left(\frac{\partial Y_{INV}}{\partial Y_{3D}}\right)^t \quad (10)$$

IV. 2D Homography Warping

When a feature is going to be measured, the estimation of the left camera position and orientation, which are obtained both from the SLAM state vector, and the normal surface patch vector are used for transforming the initial image template appearance (due to changes in viewpoint) by warping the initial template using a 2D homography. Our approach is related to the previous works of [11], [12].

Considering two camera centered coordinate systems, the transformation between two generic coordinate systems X_1 and X_2 is defined by:

$$X_2 = R\cdot X_1 + T \quad (11)$$

where R and T are the rotation matrix and the traslation vector encoding the relative position of the two coordinate systems. If X_1 is a point on the plane defined by eq. 12:

$$\pi : a\cdot x_1 + b\cdot y_1 + c\cdot z_1 + 1 = 0 \quad (12)$$

This is a plane which does not pass through the origin, and $n = (a,b,c)^t$ is the plane normal. According to this, the following relationship can be found:

$$n^t\cdot X_1 = -1 \quad (13)$$

Using the previous equation, eq. 11 can be expressed as follows:

$$X_2 = R\cdot X_1 - T\cdot n^t\cdot X_1 = \left(R - T\cdot n^t\right)\cdot X_1 \quad (14)$$

And therefore, image positions in the two camera frames are related by the 2D homography:

$$U_2 = C_2\cdot\left(R - T\cdot n^t\right)\cdot C_1^{-1}\cdot U_1 \quad (15)$$

Fig. 3 depicts the stereo geometry, and also the problems of obtaining the plane normal vector and the 2D homography for warping the initial image template using information from both cameras.

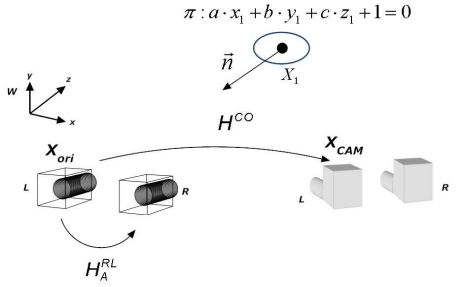


Fig. 3. Stereo geometry and locally planar surfaces

Eq. 16 denotes the relationship between the left camera and the right camera coordinate systems:

$$U_R = C_R\cdot\left(R^{RL} - T^{RL}\cdot n^t\right)\cdot C_L^{-1}\cdot U_L \quad (16)$$

The previous equation depends on the rotation matrix R^{RL} and the translation vector T^{RL} between both cameras. The values of these matrixes are known accurately, since they are estimated in a previous stereo calibration process. Supposing an affine transformation between left

and right image patches, the affine transformation H_A^{RL} can be expressed as:

$$\mathbf{H_A^{RL}} \;=\; C_R \cdot \left(R^{RL} - T^{RL} \cdot n^t\right) \cdot C_L^{-1} \qquad (17)$$

This affine transformation can be computed easily by means of 3 correspondences of non collinear points and with the assumption of locally planar patches. As it can be observed, eq. 17 depends on the plane normal vector n. From eq. 17 the product $T^{RL} \cdot n^t$ can be isolated. Denoting this product as X, it can be obtained as follows:

$$\mathbf{X} \;=\; T^{RL} \cdot n^t \;=\; R^{RL} - C_R^{-1} \cdot H_A^{RL} \cdot C_L \qquad (18)$$

All the parameters of eq. 18 are known, since the affine transformation H_A^{RL} has been previously computed, and the rest of implied matrixes are known from the stereo calibration process. According to this, a system of 9 equations and 3 unknowns, which are the components of the plane normal vector, can be found:

$$\begin{cases} n_x = \frac{X_{11}}{T_x} & n_x = \frac{X_{21}}{T_y} & n_x = \frac{X_{31}}{T_z} \\ n_y = \frac{X_{12}}{T_x} & n_y = \frac{X_{22}}{T_y} & n_y = \frac{X_{32}}{T_z} \\ n_z = \frac{X_{13}}{T_x} & n_z = \frac{X_{23}}{T_y} & n_z = \frac{X_{33}}{T_z} \end{cases} \qquad (19)$$

At the moment of a feature initialization, the plane normal vector is computed in the way it has been explained. Once this normal vector is estimated, the 2D homography between two different viewpoints can be determined using the estimation of the current left camera position and orientation and the left camera position and orientation: from the feature initialization viewpoint:

$$U_{CAM} \;=\; C_L \cdot \left(R^{CO} - T^{CO} \cdot n^t\right) \cdot C_L^{-1} \cdot U_{ORI} \qquad (20)$$

where R^{CO} and T^{CO} are the rotation and translation matrixes between the current left camera position and the reference position when the feature was initialized.

V. Experiments in Outdoor Environments

In order to test the system performance, lots of outdoor sequences in urban environment under real traffic condictions have been tested. In this work, we present only the results of two of them. The cameras used were the Unibrain Fire-i IEEE1394 modules with a baseline of 30 *cm*. Image resolution was 320×240 pixels and the images were B&W sequences. The acquisition frame rate was 30 frames per second. The sequences were processed on a laptop with an Intel Core 2 Duo processor at 2.4GHz. Camera calibration is done in a previous setup process. The Visual SLAM algorithm is implemented in C/C++ and works in real-time as long as the number of features doesn't exceed 150 approximately.

Figures 5(a) and 5(b) illustrate the aerial views of the trajectory done by the vehicle in each of the sequences. For each of the sequences two different simulations were done: without inverse depth parametrization (only 3D parametrization) and considering both parametrizations (inverse depth and 3D) with a depth threshold of 30 m.

The final map and trajectory for the first and second sequences are displayed in Fig. 6 and Fig. 7 respectively, considering the different cases. Table I shows the results of the comparison between the different experiments. The meaning of the parameters of this table are:

- % Inverse Features: Is the percentage of the total number of features in the map that were initialized with an inverse depth parametrization.
- Estimated Length (m): Is the estimate of the total distance covered by the vehicle in the sequence.
- Mean P_{YY} Trace: Is the mean trace of the covariance matrix P_{YY} for each of the features that compose the final map. This parameter is indicative of the uncertainty of the features, i.e. the quality of the map.

In the first experiment, the car starts turning slightly right and then left until the car reaches an almost straight path for approximately 100 m. Then, the car turns right until the end of the street. The estimated length run of the first sequence is 166.07 m. In the second experiment the car starts turning left and then approaches a straight path for a while. After that, the car does a sharp right turn and moves straight during some meters, yielding an estimated length run of 216.33 m.

In figures 6(a) and 6(b) the two different trajectories and maps for the first sequence are displayed in a 2D view. In the same way, figures 7(a) and 7(b) depict the two different trajectories and maps for the second sequence.

The maps are composed of the 3D position of the features with its respective covariance, which has an elliptical shape. This covariance is an indicative of the quality of the map and the uncertainty in the estimate of the 3D position of the feature in the global map. The main result that can be obtained from Tab. I or just observing figures 6(b) and 7(b) is that the uncertainty in the 3D position of the features is much lower in the cases where an inverse depth parametrization is used. This is because as mentioned previously, the uncertainty of a far feature is much lower if it is parameterized as an inverse depth feature.

The estimated trajectory reflects well the exact shape of the real trajectory executed by the vehicle in both experiments. The trajectory for the first sequence is quite similar in both of the experiments. The estimated length is also similar in both experiments, the estimated length considering inverse depth parametrization is a little bit lower than the other case. However, in the second sequence the result considering an inverse depth parametrization reflects better the shape of the real trajectory and also the estimated length is closer to the ground truth. At the end of each experiment it can be observed that the quality of the trajectory is worst than at the beginning of the sequence, which is also reflected

in the final estimated length of the trajectory. This is because at the end of the sequences the number of landmarks in the EKF filter is so high (more than 300) that provokes inconsistency in the filter. This inconsistency in the EKF is due to the errors in the approximation of the observation model by a linearization, and also because the representation of the uncertanties and 3D feature position in a common global frame. Although this is not the purpose of this work, this problem can be solved by re-linearizing the filter after some error has been accumulated creating a new submap with a local coordinate frame, expressing the uncertainties and relative 3D positions according to this new local frame.

The main drawback of the inverse depth parametrization is the computational overload of representing a feature by 6 parameters instead of 3. This drawback can be important if real-time constraints are needed for the computation of each submap. Fig. 4 depicts the state vector size during some frames of the sequence 1, considering the two experiments. As it can be observed, the difference in size due to the overload of using an inverse depth parametrization is very significant as long as new features are added to the map.

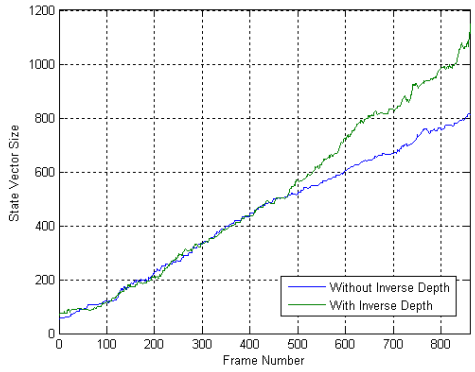


Fig. 4. Comparison of state vector size

VI. Conclusions and Future Works

In this paper we have presented a Visual SLAM approach that can estimate accurately the vehicle motion trajectory in urban roads considering small environments. In the same way a sparse map of high quality features is obtained. The system combines both bearing and depth information by means of two different types of feature parametrization: inverse depth and 3D. Inverse depth features can be switched efficiently to 3D features when its depth is below a depth threshold, reducing the uncertainty of the 3D position of far features in the global map, yielding a better localization.

We are very interested in studying the use of a dynamic threshold as a function of the kind of environment, instead of the static one that is currently used, so as to mantain the same map quality keeping real time constraints.

Considering 2D image templates and the normal vector of the plane that contains the point in the space improves the tracking considerably and it is better than using just 2D image templates. However, since the normal vector is only estimated once per feature (at the moment each feature is initialized), an update of the patch normals estimation would likely be of benefit.

In further works, a high level SLAM will be developed for mapping indoor and outdoor large environments fusing the infromation from our metric submaps. In addition, we are interested in fusing the stereo system with a commercial GPS for outdoor experiments in order to make the localization and mapping more robust, and compare our results with an accurate ground truth. In the same way, we will compare our Visual SLAM system with another techniques such as stereo Visual Odometry.

Acknowledgment

This work was supported in part by the Spanish Ministry of Education and Science (MEC) under grant TRA2005-08529-C02 (MOVICON Project) and grant PSE-370100-2007-2 (CABINTEC Project) as well as by the Community of Madrid under grant CM: S-0505/DPI/000176 (RoboCity2030 Project).

References

[1] G. Klein and D. Murray, "Parallel tracking and mapping for small AR workspaces," *IEEE and ACM Intl. Sym. on Mixed and Augmented Reality (ISMAR)*, 2007.

[2] P. Mountney, D. Stoyanov, A. J. Davison, and G. Z. Yang, "Simultaneous stereoscope localization and soft-tissue mapping for minimally invasive surgery," *Medical Image Computing and Computer Assisted Intervention (MICCAI)*, 2006.

[3] D. Schleicher, L. M. Bergasa, R. Barea, E. López, M. Ocaña, and J. Nuevo, "Real-time wide-angle stereo visual slam on large environments using sift features correction," *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2007.

[4] D. Schleicher, L. M. Bergasa, R. Barea, E. López, and M. Ocaña, "Real-time simultaneous localization and mapping with a wide-angle stereo camera and adaptive patches," *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2006.

[5] F. Dellaert and M. Kaess, "Square Root SAM: Simultaneous localization and mapping via square root information smoothing," *Intl. J. of Robotics Research*, vol. 25, no. 12, pp. 1181–1203, Dec 2006.

[6] L. M. Paz, J. Guivant, J. D. Tardós, and J. Neira, "Data association in O(n) for divide and conquer SLAM," *Robotics: Science and Systems (RSS)*, 2007.

[7] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse, "Monoslam: Real-time single camera slam," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 29, no. 6, 2007.

[8] L. M. Paz, P. Piniés, J. D. Tardós, and J. Neira, "Large scale 6DOF SLAM with stereo-in-hand," *IEEE Trans. Robotics*, vol. 24, no. 5, 2008.

[9] J. Civera, A. J. Davison, and J. M. Montiel, "Inverse depth parametrization for monocular slam," *IEEE Trans. Robotics*, 2008.

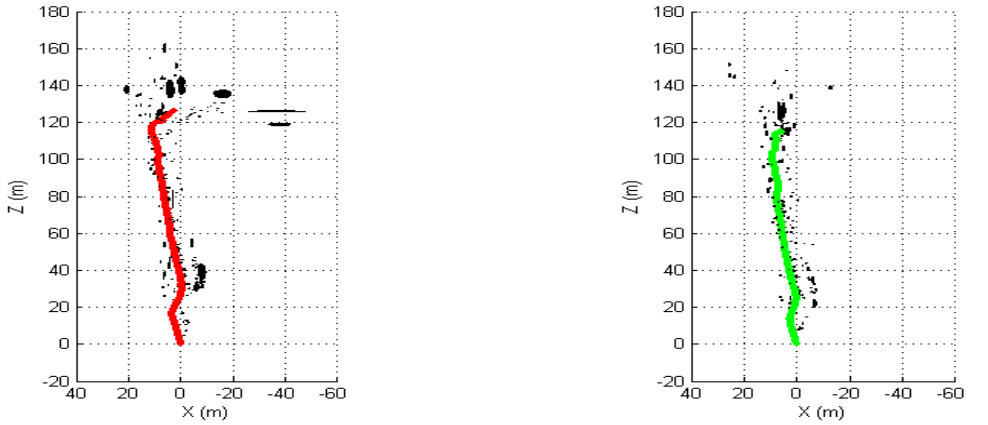
[10] C. Harris and M. Stephens, "A combined corner and edge detector," in *Proc. Fourth Alvey Vision Conference*, 1988, pp. 147–151.

[11] B. Liang and N. Pears, "Visual navigation using planar homographies," *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2002.

[12] N. Molton, A. J. Davison, and I. Reid, "Locally planar patch features for real-time structure from motion," *British Machine Vision Conf. (BMVC)*, 2004.

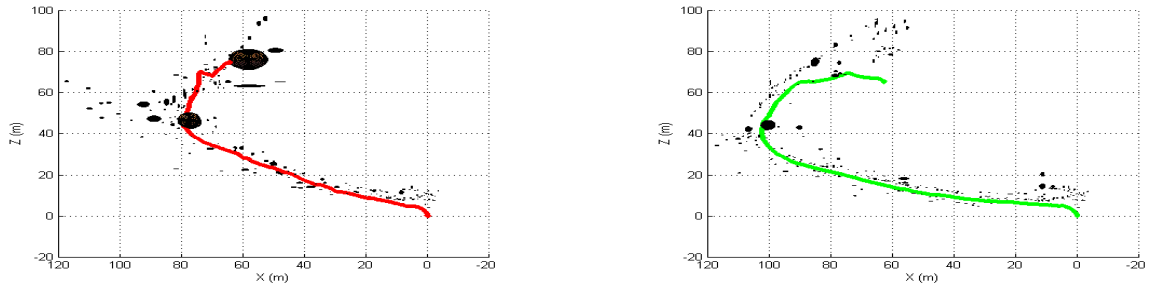
(a) Sequence 1 (b) Sequence 2

Fig. 5. Trajectories in the city for experiments 1 and 2



(a) Without Inverse Depth Par. (b) With Inverse Depth P. Z = 30 m

Fig. 6. Inverse Depth and 3D comparison: Sequence 1



(a) Without Inverse Depth Par. (b) With Inverse Depth P. Z = 30 m

Fig. 7. Inverse Depth and 3D comparison: Sequence 2

Seq.	**Case**	**% Inverse Features**	**Estimated Length (m)**	**Mean P_{YY} Trace**
1	Without Inverse Par.	0.00	133.97	2.4414
1	With Inverse Par., $Z_t = 30\ m$	12.25	129.08	0.7177
2	Without Inverse Par.	0.00	130.61	2.9729
2	With Inverse Par., $Z_t = 30\ m$	14.85	177.87	0.2188

TABLE I

Inverse Depth and 3D comparison: Estimated Length Run and Features Uncertainty