Monocular SLAM with Inverse Scaling Parametrization

D. Marzorati[2], M. Matteucci[1], D. Migliore[1], and D. G. Sorrenti[2]
[1] Dept. Electronics and Information, Politecnico di Milano
[2] Dept. Informatica, Sistem. e Com., Università di Milano-Bicocca
[matteucci, migliore]@elet.polimi.it
[marzorati, sorrenti]@disco.unimib.it

Abstract

The recent literature has shown that it is possible to solve the monocular Simultaneous Localization And Mapping using both undelayed features initialization and an Extedend Kalman Filter. The key concept, to achieve this result, was the introduction of a new parametrization called Unified Inverse Depth that produces measurements equations with a high degree of linearity and allows an efficient and accurate modeling of uncertainties. In this paper we present a monocular EKF SLAM filter based on an alternative parametrization, i.e., the Inverse Scaling Parametrization, characterized by a reduced number of parameters, a more linear measurement model, and a better modeling of features uncertainty for both low and high parallax features. Experiments in simulation demonstrate that the use of the Inverse Scaling solution improves the monocular EKF SLAM filter when compared with the Unified Inverse Depth approach, while experiments on real data show the system working as well.

1 Introduction

Recent works in Simultaneous Localization And Mapping (SLAM) have presented interesting results using a monocular camera; a simple and low power sensor that allows to estimate the bearing of interest points extracted from an image and, by means of camera motion and triangulation, the whole 3D structure of the environment [4]. Several issues affect this approach to SLAM and among them a preminent one is represented by the initialization and uncertainty modeling of the 3D elements in the map since, from a single frame, we can not estimate their real depth. Moreover, the data are affected by uncertainties that strongly depend on the observer-to-feature distance and this should be taken into account when modeling feature uncertainty too.

In their first work, Davison et al. [4], by using an extended Kalman filter to perform a real-time 6 DoF SLAM, did overcome these drawbacks by adopting a non parametric approach to the initialization of the feature depth; they also limited the scene to a maximum feature depth of about 5m. In their work the depth of an observed feature is first estimated using a particle filter, then the feature, once its distribution is close to normal, is used in a EKF-based SLAM filter. Unfortunately, this delayed use can cause a loss of information; in fact having the landmark in the map, even without knowing its distance,

BMVC 2008 doi:10.5244/C.22.94

allows immediate use of pure bearing information. To avoid this delay and to exploit low-parallax features, Solà et al.[7] presented another non parametric method that maintain several depth hypotheses combined with a Gaussian Sum Filter to cover the distribution along the whole ray to the feature, but this increases the number of hypothesis that need to be removed from the filter at some point.

An alternative solution to overcome both delayed feature initialization and multiple depth uncertainty hypothesis was introduced by Montiel et al. [6]; they showed how the use of an inverse depth parametrization makes the observation model nearly linear (at least for small camera displacements), and reduces at the same time non-Gaussianity of depth measurement. In this way, it is possible to model parameters uncertainty as Gaussian and use them in the EKF filter, without delay and reduced linearization. Another solution was proposed by Eade and Drummond [5], mapping image points, through calibrated camera model, to yield points in the image plane, w.r.t. the current pose. They introduce a linearization, i.e., the ratio with z, and do not take into account the projection uncertainties, which heavily affect the feature uncertainty. Lastly, their solution is dependent on the specific GraphSLAM approach, and on the usage of the Information form.

In this paper we present a monocular EKF SLAM filter based on a novel parametrization, i.e., Inverse Scaling, alternative to the Unified Inverse Depth, that allows undelayed use of features as well and provides a better representation of the uncertainty in features depth estimate for both low and high parallax features. Moreover, Inverse Scaling requires less parameters to represent a feature and provides a measurement model more linear w.r.t. Unified Inverse Depth, improving the EKF effectiveness.

The next section introduces the Inverse Scaling parametrization, starting from the description of the Unified Inverse Depth. In the following section the implementation of monocular EKF SLAM with Inverse Scaling parametrization is presented. In Section 4 we validate our proposal on simulated data comparing the results with the solution presented in [6] and on real data, to verify the full 6 DoF implementation of the system in a real setup.

2 The Inverse Scaling Parametrization

As proposed by Montiel et al. [6], it is possible to improve the performance of a monocular EKF SLAM adopting an inverse depth parametrization and thus allowing not only an undelayed initialization of features, but also a non-linearity reduction of the observation model. The latter result can be confirmed by analyzing the linearity of a simplified measurement equation, as showed in [2].

In the Unified Inverse Depth parametrization a 3D scene point $\mathbf{y}_i$ is defined by a vector:

$$\mathbf{y}_i = (C_{i_x}, C_{i_y}, C_{i_z}, \vartheta_i, \varphi_i, \rho_i)^T, \tag{1}$$

which represents a 3D point located at:

$$\begin{pmatrix} C_{i_x} \\ C_{i_y} \\ C_{i_z} \end{pmatrix} + \frac{1}{\rho_i}\mathbf{m}(\vartheta_i, \varphi_i) \tag{2}$$

where $(C_{i_x}, C_{i_y}, C_{i_z})$ is the camera position, i.e., the position of its projection center, when the 3D point was first observed; ϑ_i and φ_i are respectively the azimuth and the elevation

(in the absolute reference frame) for the line

$$\mathbf{m}(\vartheta_i, \varphi_i) = (cos(\varphi_i)sin(\vartheta_i), -sin(\varphi_i), cos(\varphi_i)cos(\vartheta_i))^T \tag{3}$$

and $\rho_i = 1/d_i$ is the inverse of the point depth along the line (see [6] for more details). Using this representation for each feature *i*, we obtain the following measurement equation:

$$\mathbf{h}_i = \mathbf{M}\left(\mathbf{R}_C^W\left(\begin{pmatrix} C_{i_x} \\ C_{i_y} \\ C_{i_z} \end{pmatrix} + \frac{1}{\rho_i}\mathbf{m}(\vartheta_i, \varphi_i) - \mathbf{r}_W^C\right)\right), \tag{4}$$

where $\mathbf{M}$ is the calibrated intrinsic projection matrix: $\mathbf{R}_C^W$ is the rotation matrix of the current camera position *C* w.r.t. world frame and $\mathbf{r}_W^C$ is the traslation vector of the world frame w.r.t. current camera position *C*. This representation requires the storage of six parameters in the state vector for each map feature. As demonstrated in [2], this implies a noticeable computing overhead when compared with the standard three Euclidean parameters encoding of a 3D point. Moreover, as demonstrated by Eade and Drummond [5], this parametrization does not linearize the model enough and always implies an underestimation of the uncertainty causing inconsistencies of the filter that lead pose and map estimation to an irreversible failure.

These considerations motivate the use of the Inverse Scaling Parametrization, that allows to reduce further on the non-linearity of the measurement equation and the number of parameters. The key idea is to represent 3D point in the scene using homogenous coordinates:

$$\begin{pmatrix} X_i \\ Y_i \\ Z_i \end{pmatrix} = \frac{1}{\omega_i}\begin{pmatrix} x_i \\ y_i \\ z_i \end{pmatrix}. \tag{5}$$

in this way we can reduce the number of parameters from six to four while preserving a proper modeling of depth uncertainty trhough the inverse scale parameter ω that can we assume normally distributed. Considering this new representation we can define a different measurement equations:

$$\mathbf{h}_i = \mathbf{M}\left(\mathbf{R}_C^W\left(\frac{1}{\omega_i}\begin{pmatrix} x_i \\ y_i \\ z_i \end{pmatrix} - \mathbf{r}_W^C\right)\right). \tag{6}$$

This parametrization allows also to remove the $\mathbf{m}(\vartheta_i, \varphi_i)$ term, reducing further on the non-linearity of the equation. An analytical linearity analysis, in comparison with Unified Inverse Depth, is not in the scope of this paper.

3 EKF with Inverse Scaling Parametrization

The parametrization proposed in the previous section has been validated as part of a SLAM system that uses an Extended Kalman Filter to jointly represent the map of the world and the robot pose. In this paper, we consider the camera pose represented by six degrees of freedom, and a sensor providing 2D data.

In this implementation we use the robocentric approach for the state of the filter. As presented in [1], this method allows to further reduce the inconsistency of the EKF approach (due to linearization) using a robot centered representation of the environment.

Moreover, in our case, it allows to semplify the initialization of new features as we can see in the following.

State representation in a EKF-based SLAM system using the robocentric approach is:

$$\mathbf{x} = \left[\begin{array}{ccccccc} \mathbf{x}_B^R & \mathbf{v}^R & \mathbf{x}_{F_1}^R & \dots & \mathbf{x}_{F_m}^R & \dots & \mathbf{x}_{F_M}^R \end{array} \right]^T \tag{7}$$

being $\mathbf{x}_B^R = [\phi, \gamma, \theta, x, y, z]^T$ the six degrees of freedom representation of the world/base reference frame, useful to recover the absolute map, $\mathbf{v}^R = \left[v_\phi, v_\gamma, v_\theta, v_x, v_y, v_z\right]^T$ is the camera velocity w.r.t the robot pose, and $\mathbf{x}_{F_m}^R = [x, y, z, \omega]^T$ is the feature Inverse Scaling Parametrization w.r.t. the camera position.

A constant linear and angular velocity is assumed and this produces, at each step, a roto-traslation $\mathbf{x}_{R_k}^{R_{k-1}}$ between the previous camera reference system (R_{k-1}) and the actual pose (R_k). Moreover, at each step we assume an additive white and zero mean Gaussian error due to an unknown acceleration factor **a** with covariance **Q**.

$$\mathbf{v}^{R_{k-1}} = \hat{\mathbf{v}}^{R_{k-1}} + \mathbf{a} \cdot \Delta t, \tag{8}$$

$$\mathbf{x}_{R_k}^{R_{k-1}} = \hat{\mathbf{v}}^{R_{k-1}} \cdot \Delta t. \tag{9}$$

The state is updatedin three steps: *prediction*, *update*, and *composition*. As proposed in [1] the composition between camera location at time $k-1$, and the camera movement at time k, is postponed to after the update step. This improves the motion estimation by using information about the observed features. For this reason the state, after the prediction step will be:

$$\mathbf{x}_{k|k-1} = \left[\begin{array}{cc} \mathbf{x}_{k-1} & \mathbf{a} \end{array} \right]^T \tag{10}$$

$$\mathbf{P}_{k|k-1} = \left[\begin{array}{cc} \mathbf{P}_{k-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{Q} \end{array} \right] \tag{11}$$

where the acceleration factor is simply concatenated to the state at time $k-1$.

The measurement equation of this filter is derived from our parametrization.

$$\mathbf{h}^{R_k} = \begin{pmatrix} h_x^{R_k} \\ h_y^{R_k} \\ h_z^{R_k} \end{pmatrix} = \mathbf{M} \left(\mathbf{x}_{R_{k-1}}^{R_k} \left(\frac{1}{\omega} \begin{pmatrix} x_{F_i}^{R_{k-1}} \\ y_{F_i}^{R_{k-1}} \\ z_{F_i}^{R_{k-1}} \end{pmatrix} \right) \right), \tag{12}$$

where **M** is the calibrated projection matrix of the camera, and **D** its covariance:

$$\mathbf{M} = \left[\begin{array}{ccc} fc_x & 0 & cc_x \\ 0 & fc_y & cc_y \\ 0 & 0 & 1 \end{array} \right], \ \mathbf{D} = \left[\begin{array}{cccc} \sigma_{fc_x}^2 & 0 & 0 & 0 \\ 0 & \sigma_{fc_y}^2 & 0 & 0 \\ 0 & 0 & \sigma_{cc_x}^2 & 0 \\ 0 & 0 & 0 & \sigma_{cc_y}^2 \end{array} \right], \tag{13}$$

$\mathbf{x}_{R_{k-1}}^{R_k}$ is the roto-traslation matrix between pose k and pose $k-1$; $\mathbf{h}^{R_k}$ is the projection of the 3D point in the camera frame, i.e., the pixel coordinates:

$$\mathbf{h}_k = \left[\begin{array}{c} h_{k_u} \\ h_{k_v} \end{array} \right] = \left[\begin{array}{c} \frac{h_x^{R_k}}{h_z^{R_k}} \\ \frac{h_y^{R_k}}{h_z^{R_k}} \end{array} \right]. \tag{14}$$

We can now extend the state covariance matrix with $\mathbf{D}$ to take the uncertainty of the projection matrix into consideration as well.

$$\mathbf{P}_{k|k-1} = \begin{bmatrix} \mathbf{P}_{k-1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{Q} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{D} \end{bmatrix}, \tag{15}$$

The classical EKF update equations give the new estimate of both the state vector $\mathbf{x}_{k|k}$ and the camera motion from pose $k-1$ to k.

$$\begin{aligned} \mathbf{S} &= \mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T + \mathbf{R}_k \\ \mathbf{K} &= \mathbf{P}_{k|k-1} \mathbf{H}_k^T \mathbf{S}^{-1} \\ \mathbf{P}_{k|k} &= \mathbf{P}_{k|k-1} - \mathbf{K}\mathbf{S}\mathbf{K}^T \\ \mathbf{x}_{k|k} &= \mathbf{x}_{k|k-1} + \mathbf{K}\left(\mathbf{z}_i - \mathbf{h}_k\right) \end{aligned} \tag{16}$$

where $\mathbf{R_k}$ is the measurement error covariance, $\mathbf{z}_i$ the observation and $\mathbf{H}_k$:

$$\mathbf{H}_k = \frac{\partial \mathbf{h}_k}{\partial \mathbf{x}_{k-1}} = \begin{bmatrix} \mathbf{0} \ldots \mathbf{H}_{v_k} \ \mathbf{0} \ldots \mathbf{H}_{F_i} \ \mathbf{0} \ldots \mathbf{H_a} \ \mathbf{H}_M \end{bmatrix}, \tag{17}$$

where $\mathbf{H}_{v_k} = \frac{\partial \mathbf{h}_k}{\partial \mathbf{v}^{R_{k-1}}}, \mathbf{H}_{F_i} = \frac{\partial \mathbf{h}_k}{\partial \mathbf{x}_{F_i}^{R_{k-1}}}, \mathbf{H_a} = \frac{\partial \mathbf{h}_k}{\partial \mathbf{a}}, \mathbf{H}_M = \frac{\partial \mathbf{h}_k}{\partial \mathbf{M}}$.

The last step, after prediction and update, is composition; this step allows to apply the improved roto-traslation $\mathbf{x}_{R_k}^{R_{k-1}}$ obtained by the step above to the whole state vector

$$\mathbf{x}_{k|k} = \begin{bmatrix} \ominus \mathbf{x}_{R_k}^{R_{k-1}} \oplus \mathbf{x}_B^{R_{k-1}} \\ \mathbf{v}^{R_k} \\ \ominus \mathbf{x}_{R_k}^{R_{k-1}} \oplus \mathbf{x}_{F_1}^{R_{k-1}} \\ \vdots \\ \ominus \mathbf{x}_{R_k}^{R_{k-1}} \oplus \mathbf{x}_{F_m}^{R_{k-1}} \end{bmatrix}, \tag{18}$$

where: $\mathbf{v}^{R_{k-1}} = \mathbf{v}^{R_{k-1}} + \mathbf{a}_k \Delta t, \mathbf{x}_{R_k}^{R_{k-1}} = \mathbf{v}^{R_{k-1}} \Delta t, \mathbf{v}^{R_k} = \mathbf{v}^{R_{k-1}}$; $\ominus$ is the inverse composition operator and $\oplus$ is the transformation composition operator. The corresponding covariance is:

$$\mathbf{P}_{k|k} = \mathbf{J}\mathbf{P}_{k|k}\mathbf{J}^T, \tag{19}$$

being

$$\mathbf{J} = \begin{bmatrix} \mathbf{J}_x & \mathbf{J}_v & \ldots & \mathbf{J}_{F_m} & \mathbf{J}_{a_k} \end{bmatrix} \tag{20}$$

and

$$\mathbf{J}_x = \frac{\partial \mathbf{x}_k}{\partial \mathbf{x}_B^{R_{k-1}}}, \mathbf{J}_v = \frac{\partial \mathbf{x}_k}{\partial \mathbf{v}_{R_k}^{R_{k-1}}}, \mathbf{J}_{F_m} = \frac{\partial \mathbf{x}_k}{\partial \mathbf{x}_{F_m}^{R_{k-1}}}, \mathbf{J}_{a_k} = \frac{\partial \mathbf{x}_k}{\partial \mathbf{a}_k}. \tag{21}$$

As we have introduced previously, the initialization of a new features is very simple since the filter is robocentric and the Inverse Scaling is used: a new feature initialization, being centered in the camera reference frame, is always made from position $[0,0,0,0,0,0]$. With Inverse Scaling, we can initialize the features with a huge uncertainty in the depth, as with Unified Inverse Deph, since it represents the direction of the interpretation ray. Moreover all information are described by Gaussian uncertainty over the parameters in Inverse Scaling as with Unified Inverse Depth.

Each feature is defined as:

$$\mathbf{x}^{init} = \begin{pmatrix} x, y, z, \omega \end{pmatrix}^T \tag{22}$$

when we obtain an observation $h = (u,v)^T$ of a new features from the camera, we initialize its parameters as:

$$\begin{pmatrix} x \\ y \\ z \\ \omega \end{pmatrix} = \begin{pmatrix} u - cc_x \\ v - cc_y \\ fc \\ \hat{\omega} \end{pmatrix} \tag{23}$$

being fc the focal length of the camera (we suppose unit aspect ratio), $[u,v]$ the 2D image point and $[cc_x, cc_y]$ the projection center of the camera. The initial value of $\hat{\omega}$ can be defined to cover the entire working range at bootstrap; for $1/\omega$ uncertainty to cover (with 96% probability) the range between some minimum distance min_d to infinite, ω needs to be in the 4% confidence interval $[0, 1/min_d]$. In our experiments, we used initial $\hat{\omega} = fc/(2 * min_d)$ and $\sigma_\omega = fc/(4 * min_d)$.

The new state covariance, after the initialization, is obtained using the image measurement error covariance $\mathbf{R}_k$, the state vector covariance $\mathbf{P}_{k|k}$, and the projection matrix covariance $\mathbf{D}$ (to keep in consideration the uncertainty on the camera parameters). It becomes:

$$\mathbf{x}^{init}_{k|k} = \begin{bmatrix} \mathbf{x}_{k|k} \\ \mathbf{x}^{init} \end{bmatrix} \tag{24}$$

$$\mathbf{P}^{init}_{k|k} = \mathbf{J} \begin{bmatrix} \mathbf{P}_{k|k} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_k & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & {\sigma_\omega}^2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{D} \end{bmatrix} \mathbf{J}^T \tag{25}$$

with:

$$\mathbf{J} = \left[\begin{array}{c|ccc} \mathbf{I} & \mathbf{0} & & \\ \hline \mathbf{0} & \frac{\partial \mathbf{x}^{init}}{\partial h} & \frac{\partial \mathbf{x}^{init}}{\partial \omega} & \frac{\partial \mathbf{x}^{init}}{\partial M} \end{array} \right]. \tag{26}$$

4 Experimental Results

In this section we present the capabilities of our system using a simulator for a monocular vision system, and some real video sequences. In the simulator, given a point in the map, and the position of the camera w.r.t. the map, we simulate the image formation on the device, as well as the uncertainty of the measurements. The motivation for using a simulated environment to test the proposed model is to have access to the ground truth and therefore to compare different methods using the same data. Moreover, in simulation we can easily use a Monte Carlo approach to produce a proper representation of the true uncertainty through exact particle triangulation. The simulated world is planar with a 1D camera, this totally suffice to prove the paper claims, while the real data use the robocentric 6 DoF implementation presented in the previous section[1].

[1] Parameters used for the simulated monocular system are: image resolution of 640 pixels at 30Hz and an uncertainty associated to the image measurements set to σ=0.5 pixels. We consider the projection matrices known altogether with their uncertainty, assumed normal; focal length of 650 pixels with an uncertainty of σ=3 pixels and projection center of 320 pixel with σ=2 pixels. For triangulation we used a camera displacement of 0.6m and an uncertainty associated to the image measurements set to σ=0.3 pixels.

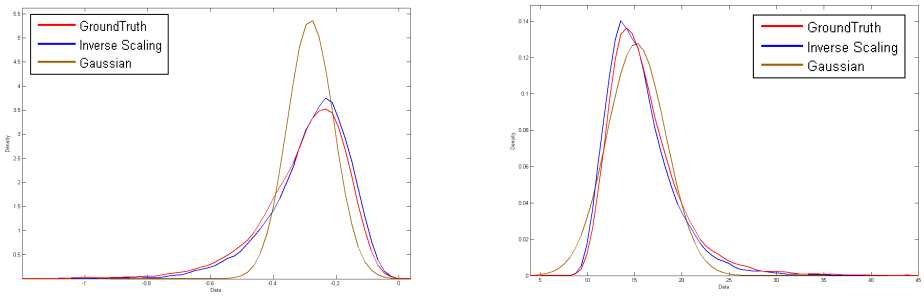

Figure 1: Estimation of 2D point at 15m away from the observer: (red) true distribution (computed with particle transformation), (blue) Inverse Scaling parametrization, (brown) classical parametrization (Gaussian distribution). The x coordinate is depicted on the left, the y on the right.

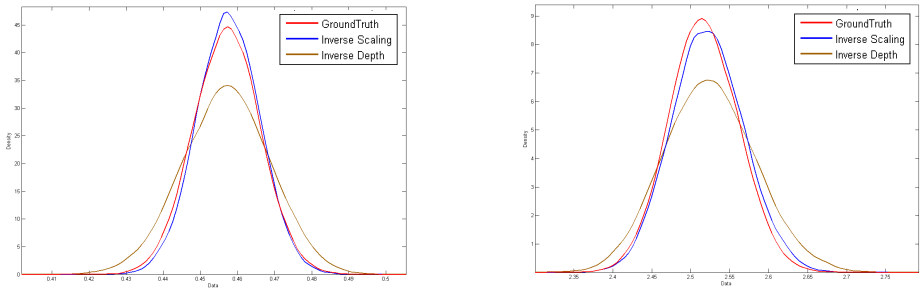

Figure 2: Estimation of 2D point at 2.5m away from the observer: (red) true distribution, (brown) Inverse Depth parametrization, (blue) Inverse Scaling parametrization. The x coordinate is depicted on the left, the y on the right.

In Figure 1 it is possible to compare the triangulation result using our model with the classical approach, i.e., Jacobian uncertainty propagation and the Cartesian $[x,y,z]^T$ point representation. The plots show the reconstruction of a scene point at 15m from the observer. We can see the non-normality of the real distribution in comparison with the classical Gaussian representation, and we confirm the better distribution approximation of the inverse scaling model with respect to classical uncertainty propagation. In Figure 2 we compare the uncertainty distribution generated using Inverse Scaling versus the Inverse Depth [6] approach when we try to estimate the 2D point at 2.5m (i.e., with a large parallax angle). The plots show that the distribution estimated by our model is realistic in this case as well and that it is more realistic with respect to the Unified Inverse Depth.

To verify if a better uncertainty modeling can lead to better SLAM results, we tested two SLAM systems in the same simulated environment where point features are equally distributed in the environment; the former implements what is proposed in Section 3, the latter uses the Unified Inverse Depth parametrization. Data association have been performed manually so that estimates are comparable and the main aspects benchmarked

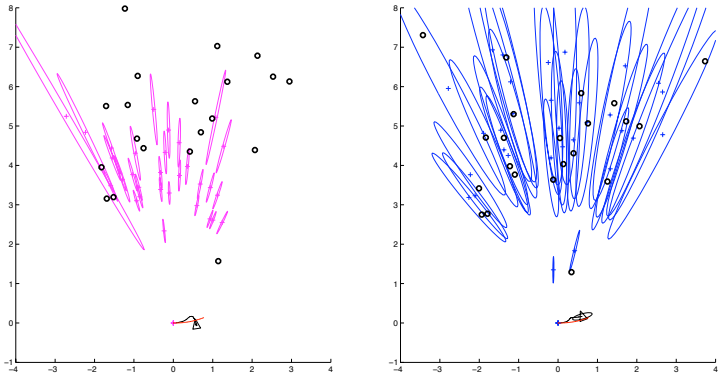

Figure 3: Map reconstruction using Unified Inverse Depth Parametrization (on the left) and Inverse Scaling Parametrization (on the right). Notice that to obtain a consistent map, in the Unified Inverse approach, we considered the absolute value of the inverse depth.

are uncertainty modeling and linearity of the measurement model. After a simple trajectory (see Figure 3) the uncertainty underestimation of Unified Inverse Scaling gives an inconsistent result; in Figure 4, we have the plot of the errors in pose estimation during the robot path, respectively for x, y and θ. As it can be noticed the variance of the robot pose estimate (the blue lines placed at $\pm 3\sigma$) is underestimated for the Inverse Depth parametrization leading to inconsistency while this is not the case for the Inverse Scaling Parametrization.

Now, we present a real application of our system in an outdoor context. In Figure 5 there are some frames taken using a 640x480 BW camera at 30Hz. The handheld camera was moved following a semi-circumference trajectory. The figure shows the map estimated using the monocular: the camera trajectory is represented in red while the features uncertainty in blue. Overlapped to the map we have shown also some images acquired by the camera with the predicted features position (in red), their uncertainty ellipses and the features matched (in blue).

5 Conclusions and Future Works

In this paper we introduce a new parametrization for monocular SLAM based on EKF filter. Compared with the Inverse Depth solution [6], our approach improves the accuracy of the uncertainty modeling, simplifies the measurement equation and reduces its non-linearity. We demonstrate this statement experimentally, using both a simulated framework to allow comparison with ground truth and a real setup.

We have developed a monocular SLAM system to show the capabilities of this new parametrization. Adopting the Robocentric approach [1] we are able to localize the camera and map the environment, reducing the underestimation of uncertainty and making the filter more robust to inconsistency. This approach will be extended using the hierarchical SLAM to map large environment and the joint compatibility test to further reduce errors

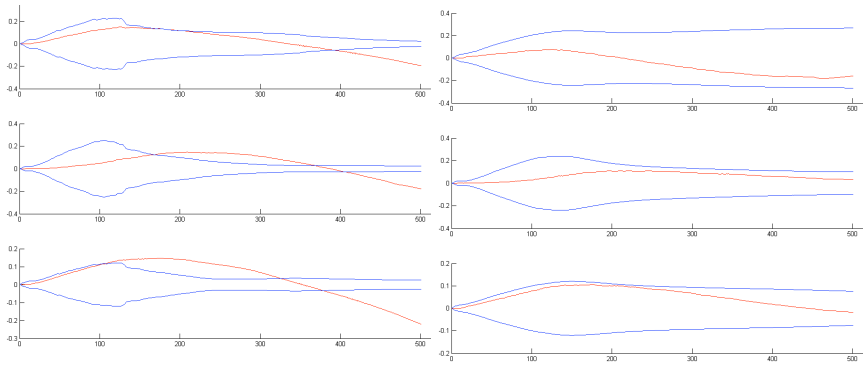

Figure 4: Error in robot localization (x,y,θ): (left) using Inverse Depth Parametrization, (right) usign Inverse Scaling Parametrization. In red the error w.r.t. the ground truth, in blue $\pm 3\sigma$

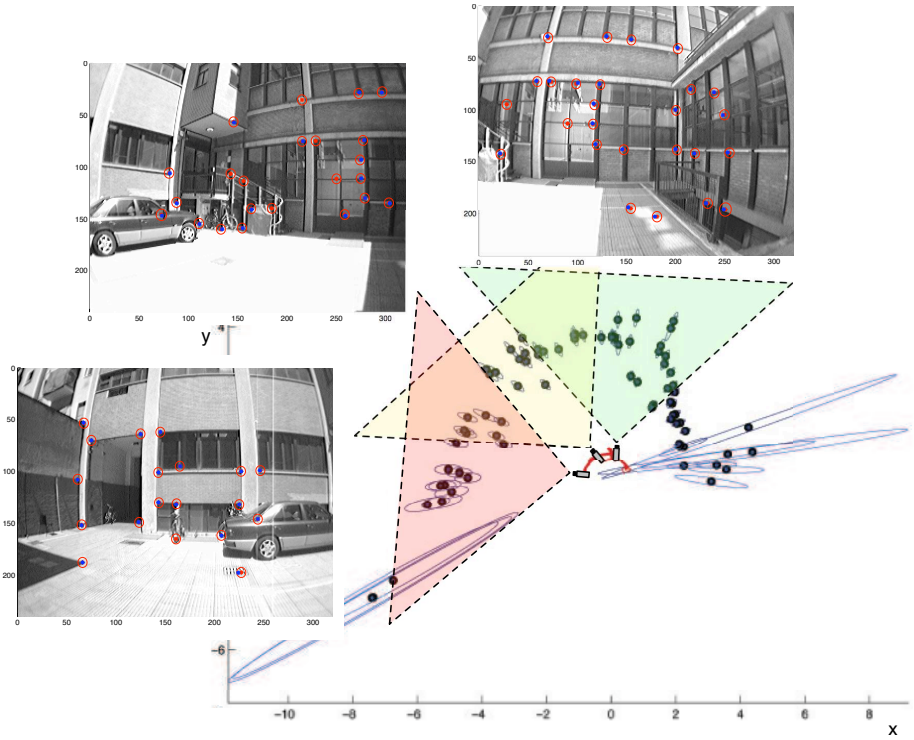

Figure 5: Map reconstruction using Inverse Scaling Parametrization in a real outdoor environment

in data association (see [3] for an example).

6 Acknowledgments

This work has been partially supported by the European Commission, Sixth Framework Programme, Information Society Technologies: Contract Number FP6-045144 (RAWSEEDS), and by Italian Institute of Technology (IIT) grant.

References

[1] J. A. Castellanos, Ruben Martinez-Cantin, J. D. Tardos, and J. Neira. Robocentric map joining: Improving the consistency of ekf-slam. *Robotics and Autonomous Systems*, 55(1):21–29, 2007.

[2] Javier Civera, Andrew J. Davison, and J. M. M. Montiel. Inverse depth to depth conversion for monocular slam. In *ICRA*, pages 2778–2783, 2007.

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

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

[5] E. Eade and T. Drummond. Monocular slam as a graph of coalesced observations. In *Proceedings of IEEE International Conference on Computer Vision 2007*, 2007 October.

[6] J. Montiel, J. Civera, and A. Davison. Unified inverse depth parametrization for monocular slam. In *Proceedings of Robotics: Science and Systems*, 2006 August.

[7] J. Solà, A. Monin, M. Devy, and T. Lemaire. Undelayed initialization in bearing only slam. In *IEEE International Conference on Intelligent Robots and Systems, IROS 2005.*, pages 2499–2504, 2005.