EKF-based SLAM fusing heterogeneous landmarks

Jorge Othón Esparza-Jiménez, Michel Devy, José-Luis Gordillo

EKF-based SLAM fusing heterogeneous landmarks

Jorge Othón Esparza-Jiménez*†, Michel Devy*, J. L. Gordillo †
*CNRS, LAAS, Univ de Toulouse, 7 avenue du colonel Roche, F-31400 Toulouse, France
Email: jorge.othon.esparza.jimenez, michel.devy@laas.fr
†Centro de Sistemas Inteligentes, Tecnológico de Monterrey, México
Email: jlgordillo@itesm.mx

***Abstract*—Visual SLAM (Simultaneous Localization and Mapping from Vision) concerns both the spatial and temporal fusion of sensory data in a map when moving a camera in an unknown environment. This paper concerns the construction of landmarks-based stochastic map, using Extended Kalman Filtering in order to fuse new observations in the map, when considering heterogeneous landmarks. It is evaluated how this combination allows to improve the accuracy both on the map and on the camera localization, depending on the parametrization selected for points and straight lines. It is analyzed using a simulated environment, so knowing perfectly the ground truth, what are the better landmark representations. Experiments on image sequences acquired from a camera mounted on a mobile robot, were already presented: it is detailed here a new front end where segment matching has been improved.**

I. INTRODUCTION

SLAM was introduced twenty-five years ago as a function required for a robot, to build a map from observations acquired by embedded sensors, while exploring an unknown environment, . It has been proposed many formal approaches to deal with the fusion of observations in the map (estimation, optimization, interval analysis...), and many representations for the environment (landmarks, grids, raw data...) [1]. While it is built, the map must allow the robot to estimate its pose; so many SLAM approaches are only devoted to build a landmarks-based map, i.e. a sparse model made with distinctive and characteristic entities located in the 3D space, which correspond without ambiguity, to some observed features.

Visual SLAM has been studied since 12 years [2], using at first an estimation framework. It could find applications not only in robotics, but also for the introduction of new services using smart devices equipped with a camera (smart phones, Kinect...). Landmarks are typically 3D points or 3D lines, that are observed by points or segments features in images; one observation does not allow to initialize such a landmark, while a 3D point is observed by an optical ray (i.e. a 3D straight line), and a 3D line is observed by an interpretation plane.

So in order to initialize such landmarks with their minimal euclidean parameters, it is necessary to wait for other observations. It is the reason why the first strategy proposed in [2] [3] applied a delayed initialization; a landmark was added in the map only when it was known in the euclidean space. This approach is unable to use landmarks that are very far from the robot. So several parametrizations have been proposed for points [4][5] or lines [6] landmarks, for their undelayed initialization, i.e. they are added in the map as soon as they are observed. Solá and al [7] have analyzed the pros and cons of several representations for 3D points and 3D lines, before they can be triangulated with a sufficient accuracy.

Here this analysis is made more complete using an heterogeneous map, where points and lines are both initialized as soon as they are detected from features extracted in images acquired by a camera moved in the environment. It is now known [8] that optimization-based methods like PTAM [9] or methods based on the g2o library [10] allow to avoid the possible divergence of methords based on estimation, due to linearization of the observation model. Nevertheless here the fusion is performed from an EKF-based SLAM method, as a very light approach that can be integrated on dedicated architecture using co-design methodology, to be used on small aerial vehicles.

In the next section it is recalled the different parametrization proposed for points and lines landmarks. Then the sections III and IV summarize the way landmarks are initialized as soon as they are observed, and then updated from next observations. Finally, It is analyzed in section V, what are the best landmarks representations when a heterogeneous map is built. Also, in the section VI, it is shown some results obtained on image sequences acquired from a mobile robot moving on a road close to buildings.

II. LANDMARK PARAMETERIZATION

In [7], Solà et al. introduce an undelayed landmark initialization (ULI) for different points and lines parameterizations. It consists on substituting the unmeasured degrees of freedon by a Gaussian prior that handles infinite uncertainty but that is still manageable by EKF.

For point and line landmark , uncertainty has distinct implications. For points, there is uncertainty in depth and it covers all the visual ray until infinity. Infinite straight lines handles uncertainty in two degrees of freedom, which correspond to a depth that should be covered up to infinity, and all possible orientations.

A. 3D point parameterizations

This section explains some point parameterizations. The aspects included in each description refer to the parameterization itself, camera projection, coordinate transformation, and back-projection.

1) Euclidean point: The parameters of an Euclidean point consist on its Cartesian coordinates.

$$\mathcal{L}_{EP} = \mathbf{p} = [x\ y\ z]^{\mathsf{T}} \in \mathbb{R}^3$$

The projection to camera frame is given by the following equation:

$$\underline{\mathbf{u}} = \mathbf{K}\mathsf{R}^{\mathsf{T}}\left(\mathbf{p} - \mathsf{T}\right) \in \mathbb{P}^2 \tag{1}$$

where,

$$\mathbf{K} = \begin{bmatrix} \alpha_u & 0 & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$

R and T are the rotation matrix and translation vector that define the camera C. Underlined vectors like $\underline{\mathbf{u}}$ represent homogeneous coordinates.

2) Homogeneous point: Homogeneous points are conformed by a 4-vector, which is composed by the 3D vector $\mathbf{m}$ and scalar ρ.

$$\mathcal{L}_{HP} = \underline{\mathbf{p}} = \begin{bmatrix} \mathbf{m} \\ \rho \end{bmatrix} = [m_x\ m_y\ m_z\ \rho]^{\mathsf{T}} \in \mathbb{P}^3 \subset \mathbb{R}^4$$

In order to convert from homogeneous to Euclidean coordinates, the following equation is applied:

$$\mathbf{p} = \frac{\mathbf{m}}{\rho}. \tag{2}$$

In the camera frame, $\mathbf{m}$ is the director vector of the optical ray, and ρ has a linear dependence to the inverse of the distance d defined from the optical center to the point.

$$\rho = \tfrac{\|\mathbf{m}\|}{d}$$

This allows to express the unbounded distance of a point along the optical ray from 0 to infinity, into this bounded interval in parameter space $\rho \in (0, \|\mathbf{m}\|/d_{min}]$.

The frame transformation of an homogeneous point is performed according to the next equation:

$$\underline{\mathbf{p}} = \mathbf{H}\underline{\mathbf{p}}^{\mathsf{C}} = \begin{bmatrix} \mathsf{R} & \mathsf{T} \\ 0 & 1 \end{bmatrix} \underline{\mathbf{p}}^{\mathsf{C}}, \tag{3}$$

where super-index C indicates the frame to which the point is refered, and matrix $\mathbf{H}$ sepecifies the frame to which the point is transformed.

The projection of a point into the image frame is performed with the following expression:

$$\underline{\mathbf{u}} = \mathbf{K}\mathsf{R}^{\mathbf{T}}\left(\mathbf{m} - \mathsf{T}\rho\right) \in \mathbb{P}^2. \tag{4}$$

Expressing an homogeneous point in the camera frame, the projected image point is $\underline{\mathbf{u}} = \mathbf{K}\mathbf{m}^{\mathsf{C}}$, and ρ^{C} is not measurable. Back-projection is then:

$$\mathbf{m}^{\mathsf{C}} = \mathbf{K}^{-1}\underline{\mathbf{u}}.$$

The complete homogeneous point parameterization is given in the following equations:

$$\mathcal{L}_{HP} = \underline{\mathbf{p}} = \begin{bmatrix} \mathbf{m} \\ \rho \end{bmatrix} = \mathbf{H} \begin{bmatrix} \mathbf{K}^{-1}\underline{\mathbf{u}} \\ \rho^{\mathsf{C}} \end{bmatrix} = \begin{bmatrix} \mathsf{R}\mathbf{K}^{-1}\underline{\mathbf{u}} + \mathsf{T}\rho^{\mathsf{C}} \\ \rho^{\mathsf{C}} \end{bmatrix} \tag{5}$$

,

where ρ^{C} must be given as prior and represents inverse-distance from the origin of coordinates.

Homogeneous point parameterization is shown in figure 1.

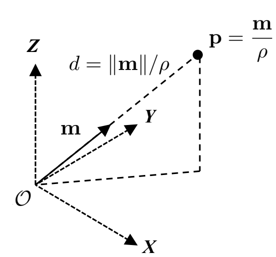

Figure 1: Homogeneous point parameterization.

3) Anchored homogeneous point: In order to improve linearity, an anchor is added as a reference to the optical center at initialization time of the landmark.

Thus, the landmark is a 6-vector that includes the anchor 3D coordinates, the Cartesian coordinates of the point with respect to the anchor, and an inverse-depth scalar.

$$\mathcal{L}_{AHP} = \begin{bmatrix} \mathbf{p}_0 \\ \mathbf{m} \\ \rho \end{bmatrix} = \left[x_0 \; y_0 \; z_0 \; m_x \; m_y \; m_z \; \rho\right]^{\mathsf{T}} \in \mathbb{R}^7.$$

The convertion from anchred homogeneous point to Euclidean coordinates can be achieved by the following equation:

$$\mathbf{p} = \mathbf{p}_0 + \frac{\mathbf{m}}{\rho}. \tag{6}$$

The projection and frame transformation process is given in the next expression:

$$\underline{\mathbf{u}} = \mathbf{K}\mathsf{R}^{\mathsf{T}}\left(\mathbf{m} - (\mathsf{T} - \mathbf{p}_0)\,\rho\right) \in \mathbb{P}^2. \tag{7}$$

The complete anchores homogeneous point parameterization is the following:

$$\mathcal{L}_{AHP} = \begin{bmatrix} \mathbf{p}_0 \\ \mathbf{m} \\ \rho \end{bmatrix} = \begin{bmatrix} \mathsf{T} \\ \mathsf{R}\mathbf{K}^{-1}\underline{\mathbf{u}} \\ \rho^{\mathsf{C}} \end{bmatrix}, \tag{8}$$

where ρ^{C} must be given as prior.

Anchored homogeneous point parameterization is shown in figure 2.

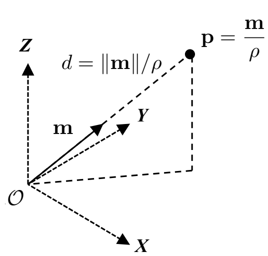

Figure 2: Anchored homogeneous point parameterization.

B. 3D line parameterizations

In this section, some line parameterizations are covered. The description of projection to image frame, bilinear transformation and back-projection are included.

1) Plücker line: A line in $\mathbb{P}^3$ defined by two points $\underline{\mathbf{a}} = [\mathbf{a}\,a]^{\mathsf{T}}$ and $\underline{\mathbf{b}} = [\mathbf{b}\,b]^{\mathsf{T}}$ can be represented as homogeneous 6-vector, known as Plücker coordinates:

$$\mathcal{L}_{PL} = \begin{bmatrix} \mathbf{n} \\ \mathbf{v} \end{bmatrix} = \left[n_x \; n_y \; n_z \; v_x \; v_y \; v_z\right]^{\mathsf{T}} \in \mathbb{P}^5 \subset \mathbb{R}^6 \; ,$$

where $\mathbf{n} = \mathbf{a} \times \mathbf{b}$, $\mathbf{n} = a\mathbf{b} - b\mathbf{a}$, $\mathbf{n}, \mathbf{v} \in \mathbb{R}^3$, and having the following Plücker constraint: $\mathbf{n}^{\mathsf{T}}\mathbf{v} = 0$.

Geometrically speaking, $\mathbf{n}$ is the vector normal to the plane π containing the line and the origin, and $\mathbf{v}$ is the director vector from $\underline{\mathbf{a}}$ to $\underline{\mathbf{b}}$. The Euclidean orthogonal distance from the line to the origin is given by $\|\mathbf{n}\|/\|\mathbf{v}\|$. Thus, $\|\mathbf{v}\|$ is the inverse-depth, analogous to ρ of homogeneous points. Plücker line geometrical representation is shown in figure 3.

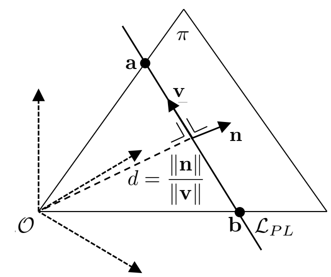

Figure 3: Plücker line geometrical representation.

Plücker coordinates transformation from camera frame is performed as shown next:

$$\mathcal{L}_{PL} = \mathcal{H} \cdot \mathcal{L}_{PL}^{\mathsf{C}} = \begin{bmatrix} \mathsf{R} & [\mathsf{T}]_{\times}\,\mathsf{R} \\ \mathbf{0} & \mathsf{R} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{n}^{\mathsf{C}} \\ \mathbf{v}^{\mathsf{C}} \end{bmatrix}.$$

The whole transformation and projection process for Plücker coordinates in terms of R , T, $\mathbf{n}$, and $\mathbf{v}$ is:

$$\mathbf{l} = \mathcal{K} \cdot \mathsf{R}^{\mathsf{T}} \cdot (\mathbf{n} - \mathsf{T} \times \mathbf{v})\,\mathcal{C} \tag{9}$$

where $\mathcal{K}$ is the instrinsic projection Plücker matrix defined as:

$$\mathcal{K} = \begin{bmatrix} \alpha_v & 0 & 0 \\ 0 & \alpha_u & 0 \\ -\alpha_v u_0 & \alpha_u v_0 & \alpha_u \alpha_v \end{bmatrix}.$$

When Plücker coordinates are expressed in camera frame, projection is only obtained by

$$\mathbf{l} = \mathcal{K} \cdot \mathbf{n}^{\mathsf{C}} \tag{10}$$

Line's range and orientation expressed in $\mathbf{v}^{\mathsf{C}}$ are not measurable.

For Plücker line back projection, vectors $\mathbf{n}^{\mathsf{C}}$ and $\mathbf{v}^{\mathsf{C}}$ are computed according to these expressions:

$$\mathbf{n}^{\mathsf{C}} = \mathcal{K}^{-1} \cdot \mathbf{l}$$

$$\mathbf{v}^{\mathsf{C}} = \beta_1 \cdot \mathbf{e}_1 + \beta_2 \cdot \mathbf{e}_2$$

where $\beta_1, \beta_2 \in \mathbb{R}$ and $\left\{\mathbf{e}_1, \mathbf{e}_2, \mathbf{n}^{\mathsf{C}}\right\}$ are mutually orthogonal.

Defining $\beta = (\beta_1, \beta_2) \in \mathbb{R}^2$, vector $\mathbf{v}^{\mathsf{C}}$ can be also expressed as:

$$\mathbf{v}^{\mathsf{C}} = \mathbf{E} \cdot \beta,$$

where $\mathbf{v}^{\mathsf{C}} \in \pi^{\mathsf{C}}$ for any value of β.

Plücker line back projection is shown in figure 4.

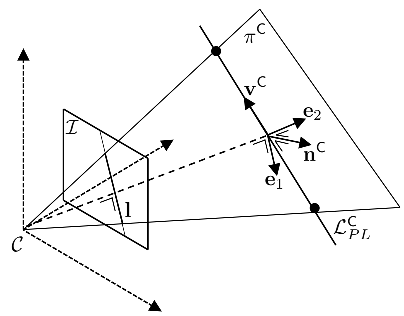

Figure 4: Plücker line back-projection.

The complete Plücker line parameterization is the following:

$$\mathcal{L}_{PL} = \mathcal{H}\begin{bmatrix} \mathbf{n}^{\mathsf{C}} \\ \mathbf{v}^{\mathsf{C}} \end{bmatrix} = \mathcal{H}\begin{bmatrix} \mathcal{K}^{-1}\mathbf{l} \\ \mathbf{E}\beta \end{bmatrix} = \begin{bmatrix} \mathsf{R}\mathcal{K}^{-1}\mathbf{l} + \mathsf{T} \times \mathsf{R}\mathbf{E}\beta \\ \mathsf{R}\mathbf{E}\beta \end{bmatrix}, \tag{11}$$

where β must be provided as a prior.

2) Anchored Homogeneous-points line: Another way of representing a line is by the endpoints that define it. Departing from the anchored homogeneous point parameterization, an homogeneous-point line is an 11-vector defined as follows:

$$\mathcal{L}_{AHPL} = \begin{bmatrix} \mathbf{p}_0 \\ \mathbf{m}_1 \\ \rho_1 \\ \mathbf{m}_2 \\ \rho_2 \end{bmatrix} \in \mathbb{R}^{11}$$

For each point, the transformation and projection of a pinhole camera is , as previously stated,

$$\underline{\mathbf{u}}_i = \mathbf{K}\mathsf{R}^{\mathsf{T}}\left(\mathbf{m}_i - (\mathsf{T} - \mathbf{p}_0)\,\rho_i\right) \tag{12}$$

An homogeneous 2D line is obtained by the cross product of two points lying on it, $\mathbf{l} = \underline{\mathbf{u}}_1 \times \underline{\mathbf{u}}_2$ and thus,

$$\mathbf{l} = \mathcal{K}\mathsf{R}^{\mathsf{T}}\left((\mathbf{m}_1 \times \mathbf{m}_2) - (\mathsf{T} - \mathbf{p}_0) \times (\rho_1\mathbf{m}_2 - \rho_2\mathbf{m}_1)\right). \tag{13}$$

Comparing this result to what was obtained for Plücker coordinates, it can be seen that the product $\mathbf{m}_1 \times \mathbf{m}_2$ is a vector orthogonal to the plane π, analogous to the Plücker sub-vector $\mathbf{n}$. Also, the term $(\rho_1\mathbf{m}_2 - \rho_2\mathbf{m}_1)$ is a vector joining the two support points of the line, therefore related to Plücker sub-vector $\mathbf{v}$.

Figure 5 shows this parameterization.

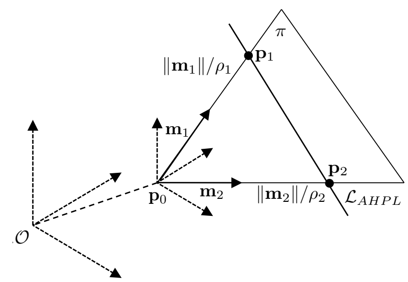

Figure 5: Anchored homogeneous-points line parameterization.

III. LANDMARK INITIALIZATION

Points are stacked as a 2-vector containing Cartesian coordinates in pixel space, and are modeled as a Gausian variable.

$$\mathbf{u} = \begin{bmatrix} u \\ v \end{bmatrix} \sim \mathcal{N}\{\bar{\mathbf{u}}, \mathbf{U}\}$$

In homogeneous coordinates,

$$\underline{\mathbf{u}} = \begin{bmatrix} \mathbf{u} \\ 1 \end{bmatrix} \sim \mathcal{N}\{\underline{\bar{\mathbf{u}}}, \underline{\mathbf{U}}\} = \mathcal{N}\left\{\begin{bmatrix} \bar{\mathbf{u}} \\ 1 \end{bmatrix}, \begin{bmatrix} \mathbf{U} & \mathbf{0} \\ \mathbf{0} & 0 \end{bmatrix}\right\}$$

In the case of lines, they can be expressed as bounded segments by means of their endpoints in a 4-vector, also with a Gaussian probability distribution.

$$\mathbf{s} = \begin{bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \end{bmatrix} \sim \mathcal{N}\{\bar{\mathbf{s}}, \mathbf{S}\} = \mathcal{N}\left\{\begin{bmatrix} \bar{\mathbf{u}}_1 \\ \bar{\mathbf{u}}_2 \end{bmatrix}, \begin{bmatrix} \mathbf{U} & \mathbf{0} \\ \mathbf{0} & \mathbf{U} \end{bmatrix}\right\}$$

The probability distribution function for infinite lines like Plücker, $pdf\mathcal{N}\left\{\bar{\mathbf{l}}, \mathbf{L}\right\}$, uses the homogeneous line representation previously shown, and the Gaussian distribution defined by :

$\bar{\mathbf{l}} = \underline{\bar{\mathbf{u}}}_1 \times \underline{\bar{\mathbf{u}}}_2$, and $\mathbf{L} = [\underline{\bar{\mathbf{u}}}_1]_\times \underline{\mathbf{U}} [\underline{\bar{\mathbf{u}}}_1]_\times^{\mathsf{T}} + [\underline{\bar{\mathbf{u}}}_2]_\times \underline{\mathbf{U}} [\underline{\bar{\mathbf{u}}}_2]_\times^{\mathsf{T}}$

The uncertainty in 3D points and lines coming from projection is kept and modeled in inverse-distance priors ρ^{C} and β^{C} through Gaussian variables. The origin of each of these priors must be inside the 2σ of the their probability density functions.

For points and point supported lines, the minimum distance must match the upper 2σ bound, thus:

$$\rho - n\sigma_\rho = 0, \quad 0 \leq n < 2$$

$$\rho + 2\sigma_\rho = 1/d_{\min}$$

Being $n = 1$, leads to,

$$\bar{\rho} = 1/3d_{\min}, \text{ and } \sigma_\rho = 1/3d_{\min}$$

The probability distribution function of a point supported line is defined as $\mathbf{t}^{\mathsf{C}} \sim \mathcal{N}\left\{\bar{\mathbf{t}}; \mathbf{T}\right\}$, with:

$$\bar{\mathbf{t}} = \begin{bmatrix} \bar{\rho} \\ \bar{\rho} \end{bmatrix}, \mathbf{T} = \begin{bmatrix} \sigma_\rho^2 & 0 \\ 0 & \sigma_\rho^2 \end{bmatrix}$$

Plücker lines prior $\beta^{\mathsf{C}} \sim \mathcal{N}\left\{\bar{\beta}; \mathbf{B}\right\}$ take the following values:

$$\bar{\beta} = \begin{bmatrix} 1/3d_{\min} \\ 0 \end{bmatrix}, \text{ and}$$

$$\mathbf{B} = \begin{bmatrix} (1/3d_{\min})^2 & 0 \\ 0 & (1/2d_{\min})^2 \end{bmatrix}$$

This penalizes lines at the back of the camera.

A. Undelayed landmark initialization algorithm

The ULI algorithm, as presented in [7] is composed by the followings steps:

1) Identify mapped magnitudes $\mathbf{x} \sim \mathcal{N}\left\{\bar{\mathbf{x}}, \mathbf{P}\right\}$.
2) Identify measurements $\mathbf{z} \sim \mathcal{N}\left\{\bar{\mathbf{z}}, \mathbf{R}\right\}$, where $\mathbf{z}$ is either point or line (i.e. $\mathbf{u}$ or $\mathbf{s}$ respectively).
3) Define Gaussian prior $\pi \sim \mathcal{N}\left\{\bar{\pi}; \Pi\right\}$ for unmeasured degree of freedom. π can either be ρ^{C}, $\mathbf{t}^{\mathsf{C}}$ or β^{C}.
4) Back-project the Gaussian measurement and get landmark mean and Jacobians.
$$\bar{\mathcal{L}} = g\left(\bar{\mathsf{C}}, \bar{\mathbf{z}}, \bar{\pi}\right)$$
$$\mathbf{G}_{\mathsf{C}} = \left.\frac{dg}{d\mathsf{C}}\right|_{\bar{\mathsf{C}}, \bar{\mathbf{z}}, \bar{\pi}}, \mathbf{G}_{\mathbf{z}} = \left.\frac{dg}{d\mathbf{z}}\right|_{\bar{\mathsf{C}}, \bar{\mathbf{z}}, \bar{\pi}},$$
$$\mathbf{G}_\pi = \left.\frac{dg}{d\pi}\right|_{\bar{\mathsf{C}}, \bar{\mathbf{z}}, \bar{\pi}}$$
5) Compute landmarks co- and cross-variances.
$$\mathbf{P}_{\mathcal{L}\mathcal{L}} = \mathbf{G}_{\mathsf{C}} \mathbf{P}_{\mathsf{CC}} \mathbf{G}_{\mathsf{C}}^{\mathsf{T}} + \mathbf{G}_{\mathbf{z}} \mathbf{R} \mathbf{G}_{\mathbf{z}}{}^{\mathsf{T}} + \mathbf{G}_\pi \Pi \mathbf{G}_\pi^{\mathsf{T}}$$
$$\mathbf{P}_{\mathcal{L}\mathbf{x}} = \mathbf{G}_{\mathsf{C}} \mathbf{P}_{\mathsf{C}\mathbf{x}} = \mathbf{G}_{\mathsf{C}} \left[\mathbf{P}_{\mathsf{CC}} \; \mathbf{P}_{\mathsf{CM}}\right]$$
6) Augment SLAM map
$$\bar{\mathbf{x}} \leftarrow \begin{bmatrix} \bar{\mathbf{x}} \\ \mathcal{L} \end{bmatrix}$$
$$\mathbf{P} \leftarrow \begin{bmatrix} \mathbf{P} & \mathbf{P}_{\mathcal{L}\mathbf{x}}^{\mathsf{T}} \\ \mathbf{P}_{\mathcal{L}\mathbf{x}} & \mathbf{P}_{\mathcal{L}\mathcal{L}} \end{bmatrix}$$

IV. Landmark update

The landmark update process starts by projecting all landmarks to the image plane, and selecting those with higher uncertainty to be corrected. In the case of points, the observation function $h()$ applies an homogeneous to Euclidean transformation $h2e()$ to the transformation + projection proceses previously presented.

$$\mathbf{z} = h2e(\underline{\mathbf{u}}) = \begin{bmatrix} u_1/u_3 \\ u_2/u_3 \end{bmatrix} \in \mathbb{R}^2$$

Innovation $\mathbf{y}$ is then computed as

$$\text{Innovation mean: } \mathbf{y} = \mathbf{z} - h\left(\bar{\mathbf{x}}\right)$$

$$\text{Innovation covariance: } \mathbf{Y} = \mathbf{R} + \mathbf{H} \cdot \mathbf{P} \cdot \mathbf{H}^{\mathsf{T}}$$

where $\mathbf{R} = \mathbf{U}$ the measurement noise covariance and Jacobian $\mathbf{H} = \left.\frac{\partial h}{\partial \mathbf{x}}\right|_{\bar{\mathbf{x}}}$.

For lines, observation function computes the orthogonal distances from the detected endpoints $\mathbf{u}_i$ to a line $\mathbf{l}$.

$$\mathbf{z} = \begin{bmatrix} \mathbf{l}^{\mathsf{T}} \cdot \underline{\mathbf{u}}_1 / \sqrt{l_1^2 + l_2^2} \\ \mathbf{l}^{\mathsf{T}} \cdot \underline{\mathbf{u}}_2 / \sqrt{l_1^2 + l_2^2} \end{bmatrix} \in \mathbb{R}^2$$

Being the EKF innovation, the difference between the actual measurement and the expectation $\mathbf{y} = \mathbf{z} - h\left(\bar{\mathbf{x}}\right)$, $\mathbf{z}$ is the orthogonal distance from the endpoints to the line defined by them. Thus

$\mathbf{y} = 0 - h\left(\bar{\mathbf{x}}\right)$

A landmark is found consistent if the squared Mahalanobis distance $MD2$ of innovation is smaller than a threshold $MD2th$

$$MD2 = \mathbf{y}^{\mathsf{T}} \cdot \mathbf{Y}^{-1} \cdot \mathbf{y} < MD2th$$

Being that the case, landmark is updated

$$\text{Kalman gain: } \mathbf{K} = \mathbf{P} \cdot \mathbf{H} \cdot \mathbf{Y}^{-1}$$

$$\text{State update: } \bar{\mathbf{x}} \leftarrow \bar{\mathbf{x}} + \mathbf{K} \cdot \mathbf{y}$$

$$\text{Covariance update: } \mathbf{P} \leftarrow \mathbf{P} - \mathbf{K} \cdot \mathbf{H} \cdot \mathbf{P}$$

V. Experimental results

This section presents the results of the simulation experiments performed for comparing the different point and lines parameterizations with heterogeneous approches that combine points and lines in the same map.

The experiments were performed in MATLAB®, departing from the EKF-SLAM toolbox [11] and adding the heterogeneous functionality.

The following parameterizations were evaluated:

- Anchored Homogenenous Point (AHP)
- Plücker Line (PL)
- Anchored Homogenenous Point Line (AHPL)
- AHP + PL
- AHP + AHPL

The environment consisted on a house conformed by 23 lines and an array of 16 points distributed equally among the walls of the house. The environment for the simulation is shown in figure 6.

The robot performs a circular trajectory of 5 m of diameter, with a pose step of 8 cm and $0.09°$. the linear noise is 0.5 cm and the angular noise $0.05°$

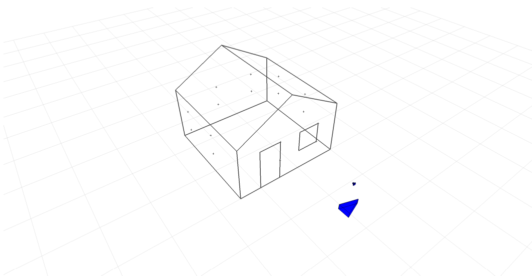

Figure 6: Environment world of the simulation experiments.

In addition to the heterogeneous landmark handling of the toolbox, two more considerations were integrated to evaluate the performance of the parameterizations in a more realistic way. The first one was the transparency of the objects in the scene. Normally, the objects in the simulaton enviroment of the toolbox are transparent, allowing to have an almost complete view of the landmarks during all the time steps. An aspect graph was implemented in order to see only the visible surfaces of the house from each camera position. Also, by default the algorithm is aware of loops, so it automatically performs feature matching, which allows to have a more accurate position estimation. It was modified in order to have the capability of initializing the landmarks as new ones on each turn and evaluate the performance in that case too. Thus, four different setup conditions were considered:

- Transparent objects with loop acquaintance.
- Transparent objects without loop acquaintance.
- Opaque objects with loop acquaintance.
- Opaque objects without loop acquaintance.

Figure 7 shows the sensor view considering transparent objects, while figure 8 shows the case with opaque objects. Figure 9 gives an example of the environment displayed after a complete turn with Plücker line + inverse depth point parameterization. It displays in green the line landmarks estimated, and in blue the point landmarks. Real, predicted, and estimated robot trajectories are displayed in blue, red, and green respectively.

A trajectory of 5 turns was performed for each parameterizaton and each condition mentioned earlier.

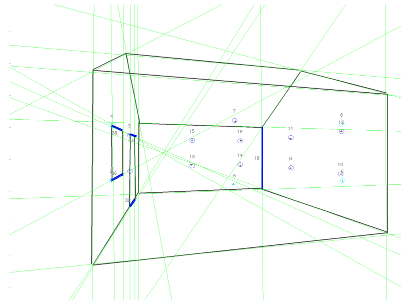

Figure 7: Sensor view considering transparent objects.

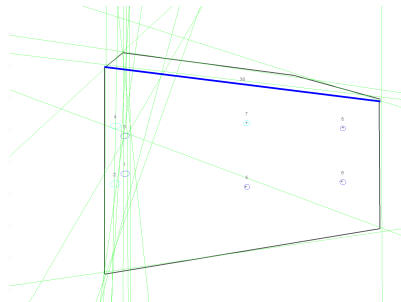

Figure 8: Sensor view considering opaque objects.

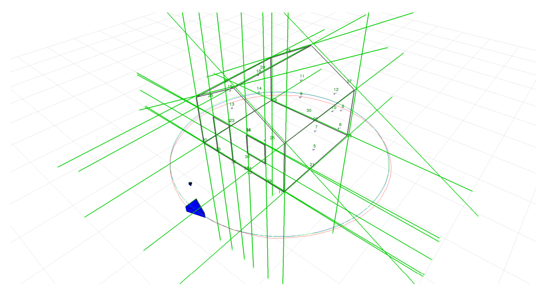

Figure 9: Image of the environment after performing a complete turn with Plücker line + inverse depth point parameterization.

The position error of the robot for each case is shown in figures 10, 11, 12, and 13.

Among the parameterizations, the highest error correspond to Plücker line. The best performance correspond to the anchored parameterizations, both for points and lines. It can be seen the improvement effect in Plücker line parameterization by the addition of points. Even for the anchored parameterizations, already having a relatively well performance while working independently, the heterogeneity brings benefits, in such a way that the combination of both AHP and AHPL is

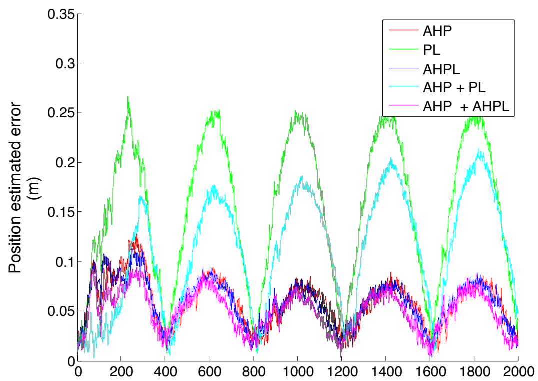

Figure 10: Transparent objects with loop acquaintance position estimation errors.

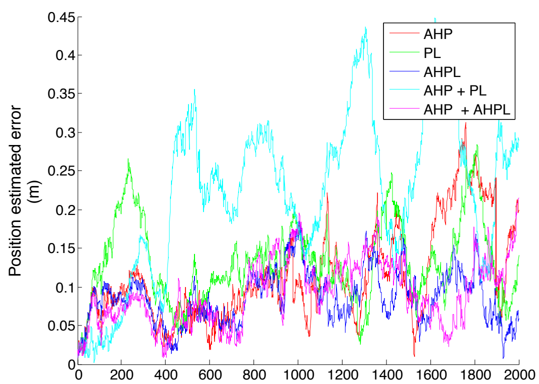

Figure 11: Transparent objects without loop acquaintance position estimation errors.

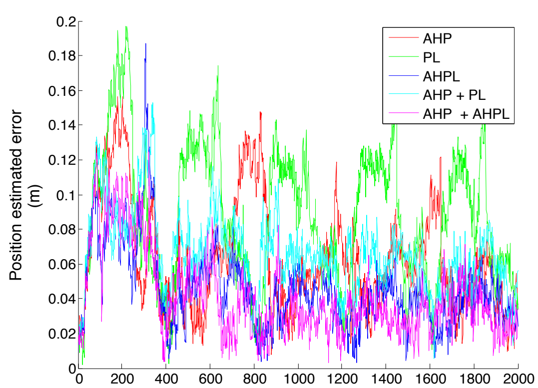

Figure 12: Opaque objects with loop acquaintance position estimation errors.

the one with the least error along the major part of the simulated trajectories.

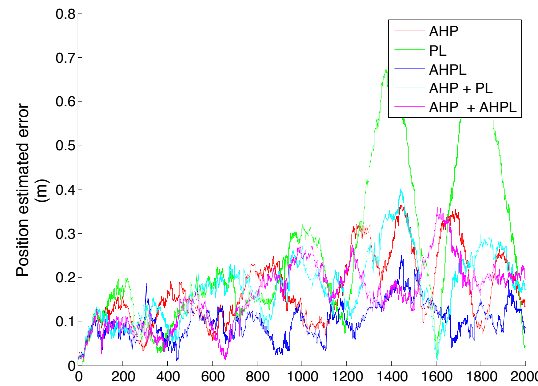

Figure 13: Opaque objects without loop acquaintance position estimation errors.

VI. Real images

The evaluation on real images is on the way; some results have been already presented, especially in [12]. Here the contribution concerns the integration of the LSD segment detector presented in [13] and a moving edge tracker based on [14] [15].

Figure 14 shows a set of frames of a sequence that have been processed for the tracking of automatically detected linear segments, and the detecton of points.

VII. Conclusion

This paper intends to prove the benefits of considering heterogeneous landmarks when building a map from an EKF-based visual SLAM method. Using only monocular vision, only partial observations of landmarks are provided by features extracted from images; here it is used undelayed initialization of landmarkslike it was proposed initially by Solaánd al [5] [6] for points and segments. It has been shown using simulated data, how the choice of the landmarks representation has an impact on the map accuracy. Finally the best ones considering the construction of map with heterogeneous landmarks, are Anchored Homogeneous Points and Anchrored Homogeneous-PointsLines.

Experiments with real images are on the way, and some preliminary results have been presented on line tracking using different sequences acquired in urban environment. In the future works, constraints will be exploited in the map, typically when points and segments are extracted from the same facades.

Acknowledgment

This work was performed during a long-term stay of J.O.Esparza Jimenez, PHd student in ITESM Monterrey, Mexico, in the robotics department of LAAS-CNRS in

France. The authors would like to thank CONACYT for the funding of this stay; this work contributes to the french FUI project AIR-COBOT.

References

[1] H. Durrant-Whyte and T. Bailey, “Simultaneous Localization and Mapping (SLAM): Part I & II,” *IEEE Robotics & Automation Magazine*, 2006.

[2] A. J. Davison, “Real-time simultaneous localisation and mapping with a single camera,” in *Proceedings of the Ninth IEEE International Conference on Computer Vision - Volume 2*, ser. ICCV ’03. Washington, DC, USA: IEEE Computer Society, 2003, pp. 1403–. [Online]. Available: http://portal.acm.org/citation.cfm?id=946247.946734

[3] T. Lemaire, S. Lacroix, and J. Sola, “A practical 3d bearing-only slam algorithm,” in *Intelligent Robots and Systems, 2005. (IROS 2005). 2005 IEEE/RSJ International Conference on*, 2005, pp. 2449 – 2454.

[4] J. M. M. Montiel, “Unified inverse depth parametrization for monocular slam,” in *Proc. Robotics: Science and Systems (RSS)*, 2006.

[5] J. Sola, A. Monin, M. Devy, and T. Lemaire, “Undelayed initialization in bearing only slam,” in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2005.

[6] J. Sola, V. C. T., and M. Devy, “Undelayed initialization of line segments in monocular slam,” in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2009.

[7] J. Sola, T. Vidal-Calleja, J. Civera, and J. M. M. Montiel, “Impact of landmark parametrization on monocular ekf-slam with points and lines,” *International journal of computer vision*, vol. 97, no. 3, pp. 339–368, 2012.

[8] H. Strasdat, J. Montiel, and A. Davison, “Real-time monocular slam: Why filter?” in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, May 2010.

[9] G. Klein and D. Murray, “Parallel tracking and mapping for small AR workspaces,” in *Proc. 6th IEEE and ACM Int. Symp. on Mixed and Augmented Reality*. IEEE Computer Society, 2007.

[10] R. Kuemmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, “g2o: A general framework for graph optimization,” in *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2011.

[11] J. Sola, D. Marquez, J. Codol, and T. Vidal-Calleja, “An EKF-SLAM toolbox for MATLAB,” http://homepages.laas.fr/jsola/JoanSola/eng/toolbox.html, 2009.

[12] T. Vidal-Calleja, C. Berger, J. Solà, and S. Lacroix, “Large scale multiple robot visual mapping with heterogeneous landmarks in semi-structured terrain,” *Robotics and Autonomous Systems*, vol. 59, no. 9, pp. 654—-674, 2011.

[13] R. Grompone von Gioi, J. Jakubowicz, J.-M. Morel, and G. Randall, “LSD: a Line Segment Detector,” *Image Processing On Line*, vol. 2, pp. 35–55, 2012.

[14] P. Bouthemy, “A maximum likelihood framework for determining moving edges,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 11, no. 5, pp. 499–511, 1989.

[15] A. I. Comport, E. Marchand, M. Pressigout, and F. Chaumette, “Real-time markerless tracking for augmented reality: the virtual visual servoing framework,” *Visualization and Computer Graphics, IEEE Transactions on*, vol. 12, no. 4, pp. 615–628, 2006.

Figure 14: Some frames of the sequence are included and the detected/tracked segments are shown in green, blue and yellow, while detected points are displayed in red.