

Simultaneous Localization and Structure Reconstruction of Mobile Robots with External Cameras

Daniel Pizarro Perez*, Enrique Santiso Gomez *, Manuel Mazo Quintas*

*University of Alcalá/Department of Electronics, Alcalá de Henares, Spain

Abstract—This paper presents a way to achieve robot positioning using visual information from cameras placed in the environment. The goal is to obtain both the global position and a not fixed amount of features from the robot. There will be defined a algorithm that implements 3D reconstruction at the same time the position of the robot is updated. The problem statement is equivalent to visual SLAM process and therefore all definitions are made in a top-down Bayesian process. This document presents a novel study of robot positioning simultaneous with 3D reconstruction, with one camera and unknown robot landmarks, which will be easily expanded to a multicamera process. It's assumed that odometric information is always present in the system, so it will be used in both estimation and initialization phases.

I. INTRODUCTION

This paper addresses the problem of using known and calibrated cameras in robot positioning tasks. These cameras form an "Intelligent Room" [3] in which some agents are able to be controlled by the environment.

Our proposal is to bring all the intelligence and complexity from the robot to the environment. Indeed in the case of vision systems it has some advantages. All the cameras can be easily calibrated, even with no human supervision and all the space requirements for visual processing hardware disappear.

Robotics agents only have to bring low level sensors for emergency tasks and a wireless link with the environment. Such external knowledge allows its interaction with other robots, as in robot cooperation or platooning, to be easier than individual interaction.

Any robot in this framework should be recognized by the environment not because of complex structure recognition, instead it will be recognized as an individual agent because of its properties of be controlled. The only requirement of the system should be this controllability of robotic agents against the environment. This work starts with the assumption that no 3D robot model is know and of course unknown metric landmarks are available on it. In the case of monocular vision it will be necessary to bring the system an accurate motion model, which is in many cases a 3D structure independent information. Although with multiple camera algorithms this can be avoided, and can be replaced with a Gaussian model, the odometric information is always present so it is not easy to discard. As is known form 'dead reckoning' solutions it could be very accurate in some short trajectory.

The method of positioning is equivalent to a SLAM [8] [5] process in the way that the position and orientation is obtained thanks to the relation of a chosen center of movement and a dynamic amount of natural landmarks which compounds the observation process. This relation is formulated in a probability framework, searching for the true a posteriori distribution which relates the position, orientation and relative position of all landmarks with the image of the last. This is the reverse problem of visual and classical SLAM [4] in which as the sensor is inside the movement model, all static objects are suitable to be used as landmark mapping.

As we will formalize later, the state vector is composed of position, orientation of the robot and several information of its structure; which is in this case the only possible group of landmarks. This novel algorithm could be placed as a method of 3D object reconstruction, so it brings 3D information of points belonging to the robot structure. Because of that it should be named Simultaneous Localization and Reconstruction (SLAR).

The paper is organized as follows: First it's presented the motion model of the system, which will be used for the estimation part of SLAR process. This model will necessary include odometry error modelling. The Section III shows the observation process of natural landmarks which will be used for correction. Once the observation process and system process is available, the SLAR process is described in section IV and V, including initialization process for one and several cameras and the recursive Bayesian inference of the probabilistic approach. In section VI several numerical results are presented as a validation process of all theory exposed. Finally conclusions and future works will close this document.

II. MOTION MODEL

In this section some definitions are presented about the motion model used as the state vector transition. As is usual in all works in statistical robot positioning, the robot model correspond with a Discrete Markov Process 'x' which evolves trough time $k \in \mathbb{N}^+$ with a transition equation f .

$$x_k = f(x_{k-1}, u_k) \quad k \in \mathbb{N}^+ \quad (1)$$

The vector x_k is composed of position r_k and orientation q_k :

$$x_k = \begin{pmatrix} r_k \\ q_k \end{pmatrix} \quad r_k \in \mathbb{R}^3, q_k \in \mathbb{R}^4 \quad (2)$$

Vector r_k represents the position $r_k = (r_{x,k} \ r_{y,k} \ r_{z,k})^T$ in the Euclidean space relative to a coordinate origin which will be established later.

Orientation vector q_k has four dimensions and is represented by a unit quaternion vector. There exist numerical advantages in using a quaternion instead of rotation matrix. Therefore it will be worth to sacrifice one dimension for this representation. As unit quaternion, the usual Euclidean norm of this vector should be always unity.

$$q_k^t \cdot q_k = 1 \quad \forall k > 0 \quad (3)$$

The orientation is also referred to one axis of the coordinate origin.

In representation of (1) vector ' u_k ' is an entry of the system at time 'k'

Additional parameters could be included in the state vector, as angular speed ' $\omega \in \mathbb{R}^3$ ' and linear speed ' $v \in \mathbb{R}^3$ '.

This vector is shared by all types of models that will be described next.

A. Non Holonomic robot model

Non Holonomics robots have a motion model in which linear velocity direction depends on orientation and so the system could be defined as follows.

$$\begin{pmatrix} r_k \\ q_k \\ \omega_k \\ v_k \end{pmatrix} = \begin{pmatrix} r_{k-1} + \Delta r_k \\ q_{k-1} \odot q((\omega_{k-1} + \Omega_k)\Delta T) \\ \omega_{k-1} + \Omega_k \\ v_{k-1} + V_k \end{pmatrix} \quad (4)$$

Now v has one dimension and represent the speed in the direction fixed by q_k .

$$\Delta r_k = v_k \Delta T \cdot (q_k)^{-1} \odot \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} \odot q_k \quad (q_k)^{-1} \odot q_k = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad (5)$$

This model consists of 11 parameters at this point. If we restrict the rotation model to a plane, as is usual in a wheeled robot, the number of parameters fall to 6. This will be the model that should be use in the results and posterior conclusions.

B. Odometric Error Model

In next sections the odometric model should be used as estimation process, and as was mentioned before it is also going to be used as a initialization process in the case of one camera.

This study is going to be focused on the wheeled structure defined by 6 parameters for a non holonomic robot moving in the plane $Z = 0$. Besides a classical differential model will be used. It's easy to identify such a robot motion with the simplified model by relating ω

and v with the angular speed of both wheels ($\omega_l \ \omega_r$) which are observed by odometric information.

$$\begin{pmatrix} \omega \\ v \end{pmatrix} = T \begin{pmatrix} \omega_l \\ \omega_r \end{pmatrix} \quad T = \begin{pmatrix} \frac{R_r}{D} & -\frac{R_r}{D} \\ R_r & R_r \end{pmatrix} \quad (6)$$

Where D is the distance between wheels and R_r the radius of each wheel.

Describing ($\omega_l \ \omega_r$) as a Gaussian process $N(\hat{\omega}_l, \hat{\omega}_r, \Sigma)$, vector ($\omega \ v$) will be also Gaussian:

$$\begin{pmatrix} \omega \\ v \end{pmatrix} = N\left(T \begin{pmatrix} \hat{\omega}_l \\ \hat{\omega}_r \end{pmatrix}, T \Sigma T^T\right) \quad (7)$$

The aim at this point is to transfer odometric known noise to states in process described in (4). Since this process is essentially non-linear it cannot be assumed that vector x_k is also Gaussian. However, as in future sections will be explained the idea is to find the additive Gaussian process which at least preserve up to second moments of the statistical distribution. Supposing that this is possible the system process across the time can be represented as:

$$x_k = f(x_{k-1}, u_k) + W_k \quad (8)$$

The entry of system ' u_k ' represents odometry information free of noise. Vector ' W_k ' represents an equivalent $N(0, \Sigma_{W_k})$ noise.

If $x_0 = N(\hat{x}_0, \Sigma_{x_0})$ then without correction the Markov Process evolve in the following way:

$$\hat{x}_k = f(x_{k-1}, u_k) \quad \Sigma_{X_k} = J_{x_k} \Sigma_{X_{k-1}} J_{x_k}^T + J_{u_k} \Sigma_{W_k} J_{u_k}^T \quad (9)$$

Where ' J_{x_k} ' and ' J_{u_k} ' represent respectively Jacobians of the non-linear process 'f'. So without correction the odometric noise is accumulative. This result is well known in the extensive literature [7] and is very important to take into account when pure estimation points are taken as entries in optimization algorithms.

III. OBSERVATION PROCESS

As is stated previously, several points from robot surface will be used to do the positioning correction at the same time these points are reconstructed. One of the most important starting conditions of this paper is to avoid artificial and known landmarks in the robot. This means that no information about robot surface is available at any starting point.

Natural features to track is a classical issue in vision processing. There are some outstanding works [1] that allow not only to track a feature in the scene, but also make a dissimilarity measure of how good is that landmark.

In this paper a simple 'Harris' [2] detector of corners is applied to the image of the robot extracted from the background. With the proper threshold a first amount of points is obtained.

The process of tracking between two adjacent frames is done with 'Mahalanobis' distance between landmark position estimation ' $y_{k|k-1}$ ' and measure ' y_k ' in the new frame 'k'.

$$d_m = (y_k - \hat{y}_{k|k-1})^T P_{y_k|y_k}^{-1} (y_k - \hat{y}_{k|k-1}) \quad (10)$$

Where ' $P_{y_k|y_k}$ ' is the estimated covariance matrix of the landmark. This distance allows to track a feature if an estimation is obtained by identification the new landmark, as the nearest ' d'_m '. To improve the 'Harris' corner detector a dissimilarity technique should be used to discard 'bad' features during the process. In [1] a dissimilarity technique is defined by using little patch of textures around the feature and obtaining an affine model that satisfies in a certain grade the transformation between frames. In this case this measure should be treated carefully, because the robot turns quickly and such a model probably will discard too many landmarks. Some spherical and cylindrical transformation models could be used instead.

IV. SIMULTANEOUS LOCALIZATION AND RECONSTRUCTION

The SLAM algorithm used in this work is equivalent to used in similar problems using robot localization with on board cameras. There is a lot of research done in vision SLAM by now and the solution to on board camera has known problems and well discussed behavior. However in our case there are a lot of new conditions that must be taken into account such as closeness of landmarks, cluttering and the limited amount of them.

The problem statement in a top Bayesian structure can be redefined eventually as follows: Let the unobserved vector of states $\{X_k; k \in \}$ be modelled as a *Markov process* of initial distribution $p(X_0)$ and a transition equation $p(X_k|X_{k-1})$. The observations $\{Y_k; k \in \}$ are assumed to be conditionally independent given the process X_k and of marginal distribution $p(Y_k|X_k)$.

All distributions correspond and are related with the process and observation transitions:

$$\begin{cases} X_k = f(X_{k-1}, W_k) \\ Y_k = h(X_k, V_k) \end{cases} \quad (11)$$

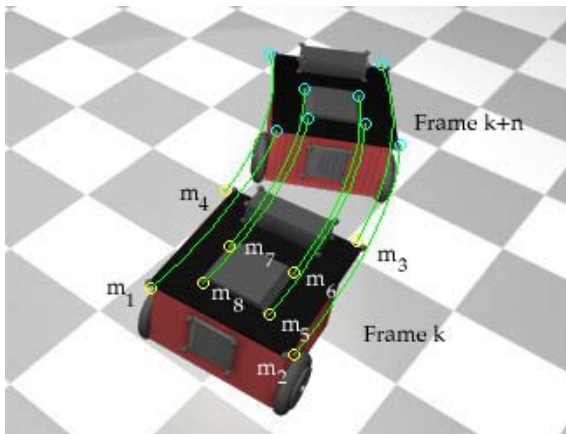


Fig. 1. Landmarks tracking over time

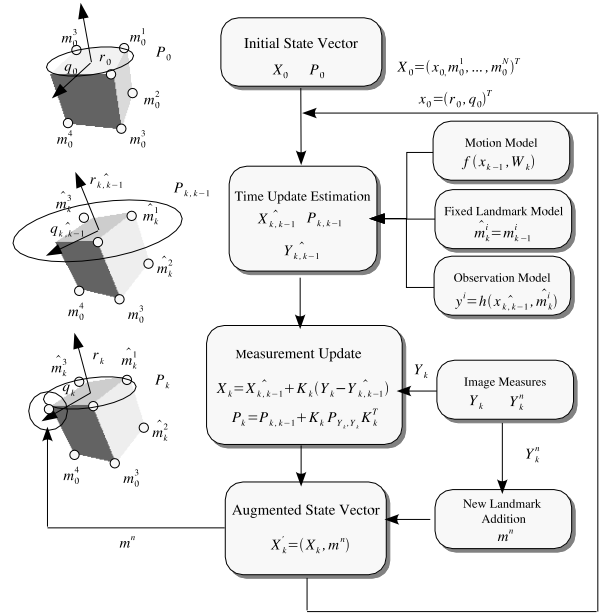


Fig. 2. SLAR Process

Where W_k and V_k are in this case two uncorrelated and mutually independent Gaussian processes which represent the uncertainty in both measures and estimation.

As all Bayesian inference problems the main goal is to obtain the true and optimal a posteriori distribution $p(X_k|Y_{1:k})$ that allows the best estimation of the process given the observations.

A. Augmented State Vector

State vector X_k is made up of two parts. The first one, namely x_k , represent the state vector described in (4) and is related with robot positioning. The second part consists of a number 'M' of known points m_k^i that belong to the robot structure and are related directly with the observation process.

$$X_k = (x_k \ m_k^1 \ m_k^2 \ \dots \ m_k^M) \quad m_k^i \in \mathbb{R}^3 \quad (12)$$

As a SLAM process, the number of landmarks should be dynamic; therefore new ones are added to the vector in $k > 0$.

Landmarks are fixed across time because represent coordinate values of points that belong to robot structure relative to robot position r_k .

The augmented state vector evolves in time as is seen in (13).

$$\begin{cases} x_k = f(x_{k-1}, W_{x_k}) \\ m_k^1 = m_{k-1}^1 + W_{m_k^1} \\ \dots \\ m_k^N = m_{k-1}^N + W_{m_k^N} \end{cases} \quad (13)$$

Usually the noise that correspond to fixed time landmarks $W_{m_k^i}$ is forced to be zero. However in some situations it should be controlled to let some landmarks move from its value over time.

B. Observation Model of Fixed Landmarks

The observation process is composed of the image of a reduced set of landmarks from which exists a component y_k of the observation vector Y_k that represents the projection of these landmarks in the image frame.

$$\mathcal{C} = \{m_k^i \mid \exists y \in Y_k, y = h(m_k^i)\}$$

Let m^i be a fixed landmark that belongs to the state vector X_k . The position of m^i in the global coordinate system of the camera, namely g^i implies using the actual position and orientation of the robot.

$$\begin{pmatrix} 0 \\ g^i \end{pmatrix} = (q_k)^{-1} \odot \begin{pmatrix} 0 \\ m^i \end{pmatrix} \odot q_k + \begin{pmatrix} 0 \\ r_k \end{pmatrix} \quad (14)$$

The projection of g^i in the camera frame finish the observation model which will be defined as the function ' $h(x_k, m^i)$ '

$$Y_k = \bar{h}(X_k) + V_k \quad \bar{h} = \begin{pmatrix} h(x_k, m^1) \\ \dots \\ h(x_k, m^M) \end{pmatrix} \quad (15)$$

C. Unscented Kalman Filter Approach

To find a solution for the Bayesian inference proposed, it will be necessary to use techniques that would be able to manage with non linear processes. Using particle filters it is possible to calculate the true a posteriori distribution with any required level of detail. However is not of our interest higher moments of the distribution or its shape. Besides particle filter is still a high computational cost solution, so it is not suitable for the entire SLAM process.

Another solution is to take the best approximation of Gaussian process and propagate through time as a Kalman Filter. To obtain the Gaussian equivalent, many authors [4] [8] use the famous Extended Kalman Filter (EKF). This algorithm has well known behavior and problems due to first order assumption which produce inconsistent estimations.

To solve the EKF inconsistency problem the 'Unscented Kalman Filter' (UKF) appears to be a good solution and is known to maintain good estimations up to second order in statistics. Unlike the EKF, the UKF does not approximate the non-linear process and observation models, it obtains an estimation of first and second order statistics sampling the true non-linear functions. The Sampling method is called *Unscented Transform* and its detailed description can be found in literature [6].

Although it is not the intention of this paper describe with detail a classic algorithm, it's convenient to establish general definition of prediction and correction steps.

- 1) *Initialization*: Using the definition of (11) the starting point consist of the statistics at $k = 0$ of all processes:

$$\begin{cases} \hat{X}_0 = E[X_0] \\ P_0 = E[(X_0 - \hat{X}_0)(X_0 - \hat{X}_0)^T] \\ Q = E[WW^T] \\ R = E[VV^T] \end{cases} \quad (16)$$

- 2) *Prediction at k*

Calculate Optimal Sampling Points (Sigma Points) and weights ' \mathcal{W} ' for each process through system dynamics and observations.

$$\begin{aligned} \mathcal{X}_{i,k-1} & \mathcal{W}_i^{\mathcal{X}} & i = 1..N_{\mathcal{X}} \\ \mathcal{Q}_{i,k-1} & \mathcal{W}_i^{\mathcal{Q}} & i = 1..N_{\mathcal{Q}} \\ \mathcal{R}_{i,k-1} & \mathcal{W}_i^{\mathcal{R}} & i = 1..N_{\mathcal{R}} \end{aligned} \quad (17)$$

Time update estimation:

$$\begin{aligned} \mathcal{X}_{k|k-1} & = f(\mathcal{X}_{k-1}, \mathcal{Q}_{k-1}) \\ \hat{X}_{k|k-1} & = \sum_{i=0}^{N_{\mathcal{X}}} \mathcal{W}_i^{\mathcal{X}} \mathcal{X}_{i,k|k-1} \\ P_{k|k-1} & = \mathcal{W}^{\mathcal{X}} [(\mathcal{X} - \hat{X})(\mathcal{X} - \hat{X})^T] \end{aligned}$$

$$\mathcal{Y}_{k|k-1} = h(\mathcal{X}_{k-1}, \mathcal{R}_{k-1}) \quad (18)$$

$$\hat{Y}_{k|k-1} = \sum_{i=0}^{N_{\mathcal{Y}}} \mathcal{W}_i^{\mathcal{Y}} \mathcal{Y}_{i,k|k-1} \quad (19)$$

3) Measurement update equations

$$\begin{aligned} P_{Y_k|Y_k} & = \mathcal{W}[(\mathcal{Y} - \hat{Y})(\mathcal{Y} - \hat{Y})^T] \\ P_{X_k|Y_k} & = \mathcal{W}[(\mathcal{X} - \hat{X})(\mathcal{Y} - \hat{Y})^T] \\ K_k & = P_{X_k|Y_k} P_{Y_k|Y_k}^{-1} \\ \hat{X}_k & = \hat{X}_{k|k-1} + K_k (Y_k - \hat{Y}_{k|k-1}) \\ P_k & = P_{k|k-1} + K_k P_{Y_k|Y_k} K_k^T \end{aligned} \quad (20)$$

For easier representation some subscripts are being omitted in covariance representations.

Finally the complete process evolve as follows:

- 1) At time $k - 1$ the robot and some landmarks are described by augmented vector X_k .
- 2) Some commands are sent to the robot to bring it to a new position at time k .
- 3) Through UKF estimation process, there are available the expected position of the image of the landmarks. Therefore $\hat{Y}_{k|k-1}$ and $P_{Y_k|Y_k}$ are used in (10) to obtain new vector of observations Y_k .
- 4) Once measures are available the UKF corrects the estimation of state vector and its covariance for the next sampling time \hat{X} and P_k .
- 5) If new landmarks are detected, they are incorporated to state statistics by any of the methods that will be described in next section.

V. INITIALIZATION PROCESSES

The process described to achieve the localization of the robot can be generalized independent of what kind of state or observation process is defined. The only actions that can not be observation model independent are the initialization processes and new landmark addition over time. This section discusses both process for single camera and multiple camera problem.

A. New Landmark Initialization with Single View

Localization process described in this paper requires an efficient and quick dynamic landmark addition to state vector. Opposite to the batch methods it will be necessary not to break the top-down Bayesian approach made in SLAM process. From UKF, position and orientation of robot are obtained, which in addition of image correspondences will triangulate the new landmark position.

To obtain a full Bayesian approach, a 'particle filtering' triangulation process similar to [4] is implemented.

- 1) From the image of a new landmark detected ' y^n ', the optical ray $X(\lambda^n)$ that passes through optical center and ' y^n ' is obtained.
- 2) A parallel state vector is initiated with ' λ_n ' as state component. The real value of ' λ^n ' must lie in the interval $(0, Z_{max})$ being Z_{max} the maximum expected distance the robot will be from camera. So the a priori statistics correspond to an uniform distribution in this interval.

$$\lambda_k^n = \lambda_{k-1}^n + W_k \quad (21)$$

- 3) From the results of UKF, for each movement of the robot, the observation function $y_k = h^n(\lambda^i, x_k)$ can be obtained.
- 4) Using a 'Bootstrap Particle Filter' and Gaussian assumption for $p(y_k|\lambda_k)$ the a posteriori distribution $p(\lambda_k|y_{k_0:k})$ approximation is obtained.
- 5) The shape of the a posteriori distribution will be sharper and more unimodal over time. When the uncertainty reaches a limit imposed by y_k variance, the landmark can be introduced in SLAM process with the covariance and mean obtained by the particle filter.

B. New Landmark Initialization with Multiple View

Initialization of a new Landmarks with multiple camera is directly made thanks to (23) and the values of all parameters from the UKF SLAM process.

Once the triangulation is made for a point ' X_i ' the new landmark coordinates for the augmented state vector are obtained by referencing the coordinates obtained to the origin of coordinates of the robot, using known position ' r_k ' and orientation ' q_k '.

$$\begin{pmatrix} 0 \\ m^i \end{pmatrix} = (q_k) \odot \begin{pmatrix} 0 \\ X^i - r_k \end{pmatrix} \odot (q_k)^{-1} \quad (22)$$

To obtain the uncertainty associated to the new landmark it's necessary to obtain joint statistics for ' m_i ' in terms of uncertainty in ' r_k ', ' q_k ' and uncertainty in the triangulation process. The new covariance matrix that will be used in SLAM can be obtained easily by using the 'Unscented Transform' into the non-linear function (22).

C. Initial State Vector: Single camera approach

Initialization process means to obtain a real knowledge about robot motion according to a prior model. It can be considered as a vision based robot calibration procedure and therefore is by itself a hard problem. Here it will be treated briefly because it's not on the focus of this paper.

Initial state vector ' X'_0 ' is defined by selected motion model and an initial amount of points from robot structure to be used as landmarks.

The proposed solution is based on obtaining an algorithm that takes advantage of information given by odometry sensors. The main idea is to fusion this information with camera projection to obtain reasonable accuracy in fewer frames than a structure recovery from correspondences approach. As odometry uncertainty grows across time, it should be necessary to design a weighted optimization algorithm.

Besides an efficient optimization algorithm, robot path during initialization is also a sensitive issue. Not all paths results in a good or even an unique solution. Conditions to reach a stable value for initialization could be enumerated:

- 1) As odometry uncertainty grows quickly across time, a minimum odometry error path is required.
- 2) Path should be a combination of angular and linear speed. Only angular or linear paths do not bring enough information for calibration.
- 3) Depending on application, robot motion will be constrained.

D. Initial State Vector: multiple camera approach

In this section the multiple camera approach for initialization process is discussed. Through point correspondence and calibrated cameras the true 3D coordinates of a point are obtained directly. Therefore to simplify all issues about multiple camera constraints, a general triangulation function is defined. For N_c cameras and a set of point correspondences ' $(y_1, ..y_{N_c})$ ' the true Euclidean coordinates are obtained.

$$X = t(y_1, ..y_{N_c}) \quad (23)$$

To assure a odometry independent algorithm a simple and very fast strategy will be a good choice to obtain motion parameters.

- 1) Making the robot turn around its axis, ' r_0 ' is easily determined by identification of all circumferences. Let $X_{i,k} \in \mathbb{R}^3$ the 3d coordinates of a point belonging to robot. According to motion, independent of k value, it should lie in a circumference of unknown radius.

For a set of N_p points the optimization algorithm should obtain the set of radius R_i for each point and the center of rotation. The cost function will be as follows:

$$\epsilon^2 = \sum_{i=1}^{N_p} \sum_{j=1}^k \frac{(X_{i,j} - r_0)^T (X_{i,j} - r_0) - R_i^2}{R_i^2} \quad (24)$$

- 2) Once ' r_0 ' is obtained, the orientation vector is trivially obtained from a straight movement

VI. SIMULATION AND NUMERICAL RESULTS

Obtaining a faithful method that can be used to test the system proposed in this paper is not an easy task. To assure a good performance indicator, it's necessary to have the real values of robot positioning in every

frame captured. Also robot landmarks should be compared to real values measured in the robot, and the real odometric noise should be modelled. We propose a different approach to validate algorithms and obtain the exact error introduced in the whole system results. Also it should be possible to use different light sources, camera configurations and robot appearance. Such an amount of advantages are reached thanks to a photorealistic 3D modelling software which is used to simulate full scene. It includes controlled light sources, camera modelling, and all the occlusion and precision effects that will be obtained using real images. Besides, the path along the robot is navigated can be easily defined by the user with a trajectory spline curve.

Using this type of simulation some experiments are made to obtain robot position and orientation. See figures (4) and (5).

VII. CONCLUSIONS

This paper has presented a novel method of robot localization by using the information taken from external vision sensors. Besides the process of localization, the 3D coordinates of points used as dynamic landmarks belonging to robot surface are obtained. To reach the goals of the paper a full Bayesian inference problem is defined (SLAR), allowing the use of powerful statistical inference techniques as 'visual SLAM' already discussed for onboard cameras problems, in the 'Intelligent Room' approach here proposed. Also, initialization process are defined for one and several cameras, which complete the main contributions of this paper.

However some hard inconveniences are not treated at detail and should be the main focus in future work. The most important are the following:

- A deep study of hard occlusion detection and modelling should be included.
- Minimum data evaluation and design of a robust process.
- Multiple camera SLAR with epipolar, trifocal and quadrifocal constraints.
- Multiple robot extension of the algorithm.

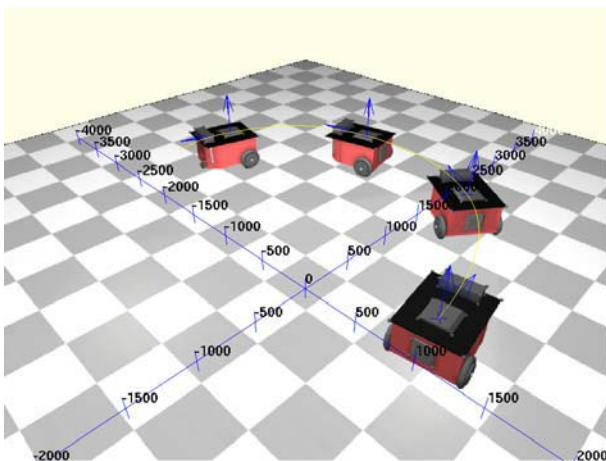


Fig. 3. Scene 3D simulation

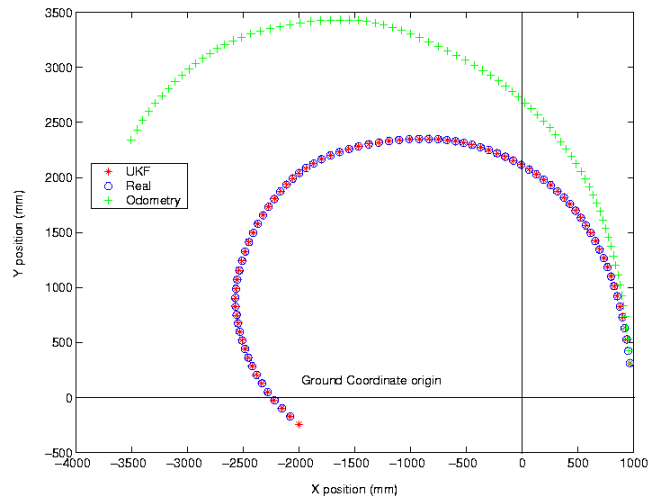


Fig. 4. Robot path in ground plane X-Y

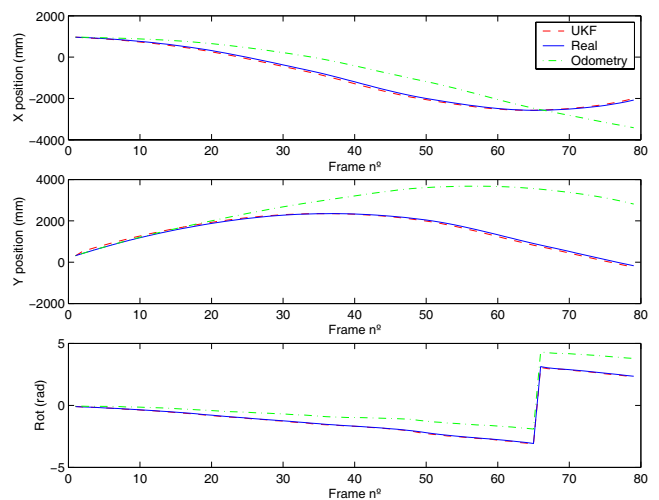


Fig. 5. Position X, Y and Rotation vs frame

REFERENCES

- [1] J. Shi and C. Tomasi, "Good Features to Track", In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 593–600, 1994.
- [2] C. G. Harris, "Geometry from Visual Motion", A.Blake and A.Yuille, editors, *Active Vision*, MIT Press, Cambridge, 1992.
- [3] Rodney Brooks, "The Intelligent Room Project", In *Proceedings of the 2nd International Cognitive Technology Conference (CT'97)*, Aizu, Japan, 1997.
- [4] A. J. Davison, "Real-Time Simultaneous Localization and Mapping with a Single Camera", In *Kroc. International Conference on Computer Vision*, Nice, oct 2003.
- [5] P. A. Beardsley, I. D. Reid and A.Zisserman and D. W. Murray, "Active Visual Navigation Using Non-Metric Structure", *ICCV5*, pp. 58–65, IEEE Computer Society Press, 1995.
- [6] R van der Merwe, A Doucet, Nando de Freitas and E Wan. "The Unscented Particle Filter", In *Advances in Neural Information Processing Systems (NIPS13)* T.K. Leen, T.G. Dietterich and V. Tresp editors. December, 2000.
- [7] J. Borenstein and L. Feng, "Measurement and Correction of Systematic Dead reckoning Errors in Mobile Robots", In *IEEE Journal of Robotics and Automation*, nov 1994.
- [8] J.E Guivant and E. M. Nebot, "Optimization of the simultaneous localization and map-building algorithm for real-time implementation". *IEEE Transactions on Robotics and Automation* vol 17.