Localization and Reconstruction of Mobile Robots Using a Camera Ring

Daniel Pizarro Manuel Mazo Enrique Santiso Marta Marron
Ignacio Fernandez

Abstract

In this paper a system capable of obtaining the 3D pose of a mobile robot using a ring of calibrated cameras attached to the environment is proposed. The system robustly tracks point fiducials in the image plane of the set of cameras generated by the robot's rigid shape in motion. Each fiducial is identified with a point belonging to a sparse 3D geometrical model of robot's structure. Such model allows direct pose estimation from image measurements and it can be easily enriched at each iteration with new points as the robot motion evolves. The process is divided in an initialization step, where the structure of the robot is obtained and an online step, which is solved using sequential Bayesian inference. The approach allows to model properly uncertainty in measurements and estimations, at the same time it serves as a regularization step in pose estimation. The proposed system is verified using simulated and real data.

Index Terms

Computer Vision, Intelligent Spaces, Robotics

I. INTRODUCTION

LOCALIZATION of mobile robots in indoor environments using a sensor network still remains to be a hot topic . The short distances involved in the localization, jointly with the

Based on "Pose and Sparse Structure of a Mobile Robot using an External Camera", by Daniel Pizarro, Manuel Mazo, Enrique Santiso and Marta Marrn which appeared in Proceedings of the IEEE International Symposium on Intelligent Signal Processing, [2007] IEEE

D. Pizarro is with the Department of Electronics, Universidad de Alcala, Alcala de Henares, 28871, Spain, Phone: (+34) 918856582, email: pizarro@depeca.uah.es
M. Mazo is with the Department of Electronics, Universidad de Alcala, Alcala de Henares, 28871, Spain,
E.Santiso is with the Department of Electronics, Universidad de Alcala, Alcala de Henares, 28871, Spain,
M. Marron is with the Department of Electronics, Universidad de Alcala, Alcala de Henares, 28871, Spain,
and I. Fernandez is with the Department of Electronics, Universidad de Alcala, Alcala de Henares, 28871, Spain,

structural elements found inside buildings, avoids in most of the cases to adapt the same radio technology that successfully made possible to partially solve outdoor localization. Instead, the special conditions of indoor localization requires different approaches as it fits better with short range sensors such as vision, ultrasounds, or recently ultra wide band (UWB) sensors.

We propose in this paper a method to retrieve the pose of a mobile robot using vision sensors that are attached to the indoor environment. The cameras form part of a sensor network known as "Intelligent Space" [1] [2] [3]. The idea behind is to place sensors in a bounded area, which are connected to a centralized system which analyze the information and make decisions. A set of "agents" such as robots, display screens or any other electronic device, are remotely controlled by the environment to accomplish a certain task. Knowing where such agents are, specially mobile robots, with enough accuracy and robustly is quite important for almost any oriented application of "Intelligent Spaces" like human assistance, robot cleaning, surveillance and more.

A. Previous Works

Despite the potential of using camera networks to localize robots, there are relatively few publications on this area compared to those in which the camera is uniquely inside the robot [4] [5]. Some examples of robot localization with camera networks can be found in the literature, where the robot is equipped with artificial landmarks, either active [6] [7] or passive ones [8] [9]. In other works a model of the robot, either geometrical or of appearance [10] [11], is learnt previously to the tracking task. In [12] and [13], the position of static and dynamic objects is obtained by multiple camera fusion inside an occupancy grid. An appearance model is used afterwards to ascertain which object is each robot. Despite the technique used for tracking, the common point of many of the proposals found in the topic comes from the fact that rich knowledge is obtained previously to the tracking, in a supervised task.

B. Localization based on Natural Appearance

In this paper we present a localization system which not necessary relies on invasive beaconing or previous supervised learning tasks. Instead of that, we propose a system that does not need artificial landmarks placed on the robot or any initially learned CAD model of its structure. The system needs only as prior information, the rigidity assumption in the geometry of the object to track and the calibration parameters from the set of cameras.

Obtaining the pose of a mobile robot using cameras, in the absence of other information, requires to define a common coordinate origin attached to the robot's volume from which to refer the pose. As a consequence, in general terms, the pose cannot be recovered without recovering also geometrical information that defines the robot's coordinate origin. In most of the cases the robot's geometry is easily observed in the images as points, lines or any other tractable entity, whose three-dimensional equivalent is possible to be inferred from image projections. As a consequence, in this paper the robot's pose is jointly obtained with a set of three-dimensional points from the robot's structure.

The computer vision community has developed a set of wide accepted solutions for the problem of obtaining rigid structure from motion. There are many publications of both sequential [5] [14] and batch approaches [15] [16] [17] and it is considered a mainly solved problem. Most of these methods are focused on scene reconstruction using a moving camera, so that the geometry completely surrounds it, instead of occupying a small amount. The main efforts are spent at the moment on the creation of unsupervised methods for reconstruction, which are able to manage with high amount of information (thousand of points in hundreds of different views) or incomplete data sets.

Usually online methods can be split up into two parts. Firstly, an unsupervised initialization algorithm is used to set up geometry from motion using a metric reference. Using auto-calibration

techniques [18] the camera parameters are obtained in the case they are unknown. The second step, which is online, combines the previous time estimation to obtain object pose given the geometry [17]. The intention of this paper is to show how to adapt such approaches to compute the pose and structure of the robot.

In [19] a system with the same objectives that the present paper, which performs robot localization using a single camera is proposed. In such proposal the initialization is solved using a "bundle adjustment" approach which needs the odometry information from the robot to serve as metric information. The online solution is solved using a robust method to avoid outliers allows the system to work under oclussions and false matchings. This paper extends the proposal made in [19] and extend it to work with several cameras, exploring the especial assumptions necessary in such case. The statistical approach is maintained in this paper as the basis to achieve the online pose and structure of the robot.

The paper is organized as follows: In §II the objectives and a general schema of the proposal is presented. The problem of measuring information with the camera is explained in §III. The initialization of pose and geometry of the robot from several cameras is presented in §IV. In §V the Online algorithm which obtains robot's pose given image measurements is explained. Finally in §VI and §VII both the experimental results and the conclusions of the paper are discussed.

II. Objectives and Proposed Approach

The objective of this paper is to obtain the pose, and consequently structure, of a mobile robot which is seen by a set of calibrated cameras fixed to the environment. The pose of the robot and the extrinsic parameters of the cameras are referred to a global coordinate origin O_W, which is set up in a calibration step. We propose a system capable of reducing the information previously required from the object to localize. Unlike the single camera solution presented in [19], where the odometry readings are necessary in the set up, the proposal presented in this paper take advantage

of using several cameras, removing the need of additional information apart from the images.

Our proposal to indoor localization consists of a series of different blocks specialized in retrieving and filtering the information available from the cameras in order to obtain the pose of the robot. The processes are divided into those which compute the pose of the robot online and those contributing to set up the information required by the online algorithm (Initialization processes). In Fig. 1 a schematic view of the entire algorithm is given

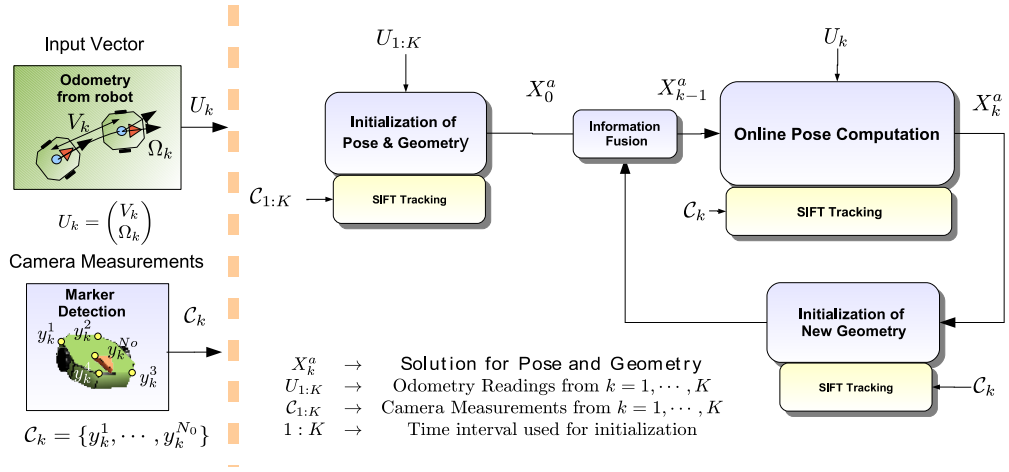


Fig. 1. Algorithm's schematic view

1. **Initialization of Pose and Geometry**: Initially, neither the pose nor the structure of the robot are known and so a method to ascertain both is proposed. A short trajectory of the robot is sufficient to get enough accuracy in pose and structure. Important issues concerning immunity against outliers, accuracy in function of robot path and number of cameras is addressed in this block.
2. **Online Pose Computation**: The online computation assumes that the initialization step was successful, so the structure and last pose are available for computing the next one. This block must be robust to avoid false matchings between the robot's structure and its measurements.
3. **Online Initialization of New Geometry**: Apart from obtaining the pose online, the system

is able to increase the knowledge about robot's from that obtained in the initialization. The goal is to use the information given at any time by the online process to include new robot's geometry points that were not previously known. This step provides a simultaneous approach for structure and motion retrieval.

4. **Measurement Process**: The measurements with the set of cameras provide the positions of the known structure points from the robot in the image plane. A method of natural marker tracking is proposed which combines an interest point detector with a tracker of points.
5. **Information fusion**: This block combines the solution to pose given by all available methods. In this proposal its inclusion is given as a matter of clarity but its implementation is not detailed.

A. Definitions and Notation Used

Robot's pose at time k is described by a vector X_k. Usually for 3D motion (6 D.O.F) the vector is composed of 3 position components and 3 orientation angles $X_k = (x_k, y_k, z_k, \alpha, \beta, \gamma)$. For wheeled robots whose motion lie on a plane vector pose X_k is reduced to 3 components (x_k, y_k, α). Motion model $X_k = f(X_{k-1}, U_k)$ obtains actual position with respect to previous time and the input U_k given by odometry (i.e. angular speed and linear speed of the robot), if available. The kind of motion model is tightly coupled with the robot's hardware and the odometry measurements. In our proposal, it is not essential to know exactly how the robot moves, but if available it can be useful to increase the convergence of the algorithms.

The geometry of the robot is composed by a sparse set of N 3D points $\mathcal{M} = \{M^1, \cdots, M^N\}$ referred always from a local coordinate origin described by robot's pose X_k. The points are static in time due to robot's rigidness, and thus, no temporal subindex is required. Function $M^i_{X_k} = t(X_k, M^i)$ uses actual pose X_k to express M^i in the global coordinate origin O_W that X_k is referred to (see Fig. 2).

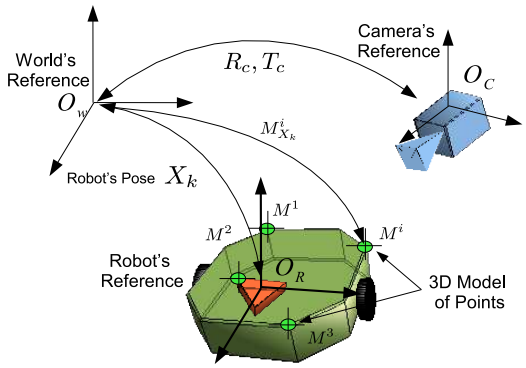


Fig. 2. Spatial relationship between world's coordinate origin O_W, robot's coordinate origin O_R and camera's coordinate origin O_C.

The augmented vector X_k^a, which is the state vector of the system, is defined as the concatenation in one column vector of both the pose X_k and the set of static structure points $\mathcal{M}$:

$$X_k^a = \left(X_k, M^1, \cdots M^N\right) \tag{1}$$

The camera sensor is modelled by a perfect "pin-hole" model described by its 3x4 projection matrix P, which encodes intrinsic and extrinsic parameters. The camera projection model is expressed by the non-homogeneous transformation $y = h(M_X, P)$ which converts a 3D point M_X expressed in a global coordinate origin O_W into its 2D projection y in the image plane using camera parameters P.

III. Measurement of Natural Markers

On most of natural objects we can find points whose image projection is able to be tracked in the image plane independently of the position the object occupies and based on local properties found in the image. (i.e lines, corners or color blobs). Those points are considered natural markers, as they serve as reference points in the image plane that can be easily relate with their three-dimensional counterparts. The set of methods focused on tracking natural markers have become a very successful and deeply studied topic in the literature [20] [21], as they represent the basic

measurements of most of existent reconstruction methods.

The process of tracking is roughly divided into two main steps: the detection of image candidates of being natural markers, and the process of their identification under different viewpoints.

A. Detection of Natural Markers

The development of stable interest point detectors has been successfully achieved since the first corner detectors where applied. The works proposed in [22] and later improved by the widely known "Harris" detector [23] have been extensively used in many vision geometry tasks. The "Harris" detector has proved to be stable enough under a variety of projective transformations, allowing its use as a reliable natural marker detector. The main drawback of the "Harris" corner comes from its sensitiveness to image scale, failing to detect a corner in objects at very different distances from the camera. In [24] a scale invariant version of the "Harris" detector was proposed, yielding to a very robust and reliable marker detector which will be used in this paper.

Given an intensity image $I(u,v)$, the multiscale Harris detector gives a number of N_o points encoded in the set $C = \begin{pmatrix} y^1 & \cdots & y^{N_o} \end{pmatrix}$ which are candidates of being points belonging to robot's structure.

B. Matching of Natural Markers

The process of matching consists of describing each fiducial included in the set C such that it can be identified in subsequent images taken at different object's poses. In the literature there is a vast knowledge on how to efficiently track fiducials [25] [26] [24], by using appearance information retrieved around the point detected (i.e texture patch).

Among them, the most used nowadays was proposed in [24] under the acronym SIFT. It has been extensively used in many tasks as Structure From Motion (SFM) or object recognition.

The SIFT method consists on finding a quasi-affine invariant descriptor d_k^i which represents

each point y_k^i. Instead of directly using image appearance, the descriptor is obtained by codifying the appearance using a set of orientation histograms from a local texture patch around the point. The orientation histograms are referred to the principal orientation found in the patch, so that the same descriptor is obtained under different transformations. A simple Euclidean distance is enough for finding a correspondence between descriptors of the same point. Each descriptor is associated to a scale detected by multiscale "Harris" and a principal orientation.

In general the SIFT method introduce false matchings when a set of descriptors are used to identify their new position in input images. Those false matchings are named "outliers" and their number or identity are not available using only image appearance (see Fig. 3).

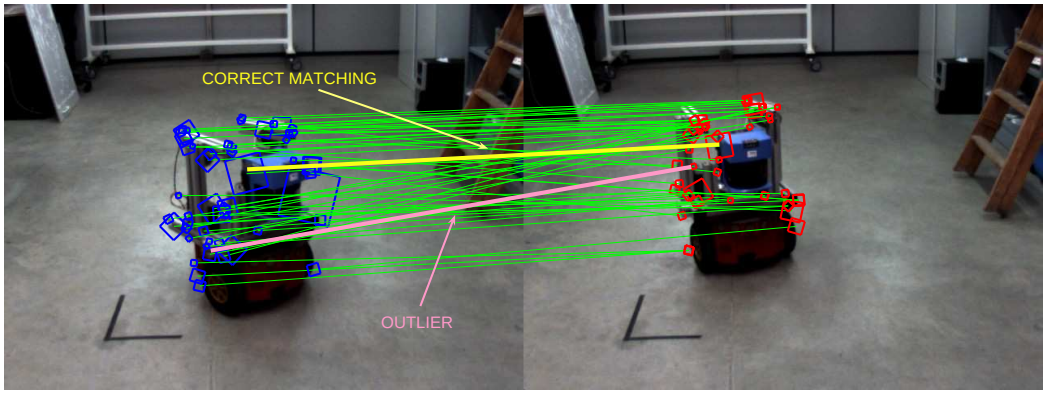


Fig. 3. Matching of natural markers, represented as fine lines, between two positions of the robot viewed in the same camera.

In the rest of the paper, the way the method SIFT is applied is left in the background with the intention of not blurring the concept. However, the measurements and their identification with the robot's structure are based on descriptor matching, an thus they are susceptible to contain outliers.

IV. Initialization of Pose and Geometry

In this section the initialization of the pose and geometry of the mobile robot by a minimum of a pair of cameras is assessed. Due to the wide-baseline configuration, where two cameras can give complementary views of the robot's geometry, we propose in this section a structure from motion

scheme of reconstruction. The basic idea is that each camera gives a series of tracks corresponding to the geometry they seen, taken from a short sequence of the robot in motion. Such information is combined to give the common solution to the robot's pose and the reconstruction of the geometry seen by all cameras. In this approach, the matching of points through several cameras is not necessary and it is replaced by tracks on each camera due to robot motion, which is generally a better posed problem.

To find the solution, we propose to use a "Bundle Adjustment" technique, which is able to optimally and efficiently reconstruct the position and structure of the robot using image measurements. The presented method obtains the solution for the set of augmented vectors along the initialization sequence $X_0^a, \cdots, X_K^a$ that minimize the error between measurements in the image plane and the expected measurements obtained by using image projection models and the structure points included in X_k^a.

A. Preliminary definitions

The initialization sequence consists of a set of K time samples starting from $k = 0$ where the robot is in motion. The pose vectors $X_0, \cdots, X_K$ define the position and orientation of the robot at each time sample of the trajectory. Whenever the odometry system is available in the robot, a noisy measurement of each $U_i \quad i = 1, \cdots, K$ gives an estimation of the pose using the recursive motion model f and the starting pose X_0.

There is a set of $N_C > 2$ cameras involved in the initialization process and, for clarity purposes, all the cameras observe the robot in the sequence. On each camera, let say camera n, a set of N_n points from the robot's structure are tracked using the SIFT method commented in III. We cannot assure that for different cameras the number of detected measurements remain constant so, in general, N_n is function of the camera index.

At frame k and with the camera n the measurement vector consist of the following:

$$Y_k^n = \begin{pmatrix} y_k^1 & \cdots & y_k^{N_n} \end{pmatrix} \quad y_k^i = (u_k^i, v_k^i), \tag{2}$$

where for clarity, the upper index of the vector Y_k^n refers to the camera index, and the one in y_k^i states for the number of detected feature, removing the reference to the camera n which is left implicit. The global measurement vector Y_L is the concatenation in a single column vector of all the measurements from all cameras.

$$Y_L = \left(\underbrace{\begin{pmatrix} Y_1^1 & \cdots & Y_K^1 \end{pmatrix}}_{\text{Camera 1}} \quad \cdots \quad \underbrace{\begin{pmatrix} Y_1^{N_c} & \cdots & Y_K^{N_c} \end{pmatrix}}_{\text{Camera } n} \right)^T \tag{3}$$

Each single measurement y_k^i at time k is a function of the parameters of the camera it belongs to, the pose vector X_k and its respective point M^j of the object's structure.

$$y_k^i = h(t(X_k, M^j), P_n) + v_k^i \quad \mathbf{v_k^i} = N(0, \Sigma_v), \tag{4}$$

where the correspondence of M^j with the ith point seen by the camera n is given by the following default ordering in the augmented vector:

$$X_k^a = \left(X_k \quad \underbrace{\begin{pmatrix} M^1 & \cdots & M^{N_1} \end{pmatrix}}_{\text{Camera 1}} \quad \cdots \quad \underbrace{\begin{pmatrix} M^r & \cdots & M^{r+N_{N_c}} \end{pmatrix}}_{\text{Camera } N_c} \right), \tag{5}$$

where $r = \sum_{n=1}^{N_c - 1} N_n$.

B. Building the Cost Function

The complete set of unknowns is composed of the set of poses in the trajectory $X_0, \cdots, X_K$ and the 3D coordinates of the points seen by the cameras $M^j, j = 1, \cdots, \sum_{n=1}^{N_c} N_n$. All parameters are packed together into the objective vector Φ:

$$\Phi = \begin{pmatrix} X_0, \cdot, X_K, M^1, \cdots, M^N \end{pmatrix} \quad N = \sum_{n=1}^{N_c} N_n \tag{6}$$

A cost function is built in function of Φ to compare the error between the real measurement of vector Y_L with its estimation, here named as $\hat{Y}_L$, and obtained using the projection models (4):

$$\epsilon^2 = (Y_L - \hat{Y}_L)^T \Sigma_L^{-1} (Y_L - \hat{Y}_L), \tag{7}$$

where Σ_L represents the covariance matrix of vector Y_L, which is a diagonal block matrix, where each block represents the noise of a single measurement Σ_v. The expression (7) can be rewritten into the following addition of terms:

$$\epsilon^2 = \sum_{k=1}^{K} \sum_{n=1}^{N_c} \sum_{i=1}^{N_n}, (y_k^i - \hat{y}_k^i) \Sigma_v^{-1} (y_k^i - \hat{y}_k^i)^T \tag{8}$$

,

In [19], a similar approach is made by using a single camera, however in that case the Σ_L models the growing error behavior of the odometry estimation, as it is included as a metric reference in the algorithm.

The minimum of (7) with respect to Φ gives the value required to reconstruct the entire initialization trajectory. To reach the minimum, the iterative optimization method Levenberg-Mardquardt is used. The analytical expression of the first and second derivatives of (7) with respect to the unknowns are required to compute a single step of the optimization.

B.1 Outliers Rejection

As the tracking used in the initialization is based on image appearance matching it is very probable the presence of erroneous measurements inside Y_L which do not correspond to any 3D point from the robot's structure. The identification of such points, called "outliers" is decisive for a successfully initialization.

The cost function (7) is designed to model possible Gaussian deviations of the solution compared to the measurements, which are suitable to contain noise. The distribution of outliers gen-

erally does not fit into such modelling and so their presence in the optimization yields to a biased solution. The solution to that situation comes from using a robust cost function capable of modelling the outlier distribution, removing its influence in the solution. The equivalent cost function results in the following:

$$\epsilon^2 = \sum_{k=1}^{K} \sum_{n=1}^{N_c} \sum_{i=1}^{N_n} \rho((y_k^i - \hat{y}_k^i)\Sigma_v^{-1}(y_k^i - \hat{y}_k^i)^T), \tag{9}$$

where $\rho(s)$ can be any increasing function with $\rho(0) = 0$ and $\frac{d}{ds}\rho(0) = 1$ (See [27] for more details). The following ρ_i is proposed in this paper, which models the outliers as a Cauchy distribution:

$$\rho(s) = b^2 \log 1 + \frac{s}{b^2}, \tag{10}$$

where b^2 is used as a control parameter which determines for which range of s the function is approximated by a quadratic function, and which range is considered as outliers.

One of the main drawbacks of using robust cost functions is the increasing of the nonlinearity of the problem, which generally affects the convergence properties of the algorithm.

B.2 Initialization before optimization

The optimization method to minimize the cost function requires a guess value for the solution Φ from which start iterating. It is preferable to chose a value as close as possible to the real solution, so that the probability to reach the global minimum increases.

A very simple proposal is made in this paper to get an initial guess of the solution. It consists of the following steps:

- For each time instant k and camera n the center of mass of the points encoded in Y_k^n is obtained. (μ_k^n).

- The position of the robot at time k x_k, y_k, z_k, not including the orientation parameters $\alpha_k, \beta_k, \gamma_k$, is obtained by camera triangulation of the N_c center of mass calculated, supposing they represent the same point in the three-dimensional space.
- Once the set of $K+1$ positions are available $x_0, y_0, z_0, \cdots x_K, y_K, z_K$, the set of orientations $\theta_0, \beta_0, \gamma_0, \cdots, \theta_K, \beta_K, \gamma_K$ are set so that they follow the kind of motion we expect in the object. Generally the objects present a non-holonomic motion, so we aligned the orientations, following the curvature of the motion. However, in the case that the object's motion is entirely holonomic, a random value can be used instead.
- The geometry of the robot $M^1, \cdots, M^N$ is initialized randomly around a volume bounded by a sphere of radius R, which can be obtained by calculating the minimum sphere that covers the image measurements of all cameras at a specific frame.

In the case that the odometry readings are available, only the initial pose X_0 and the geometry of the robot are guessed as was mentioned before. After their estimation the motion model and the odometry readings $U_1, \cdots, U_K$ generate the rest of poses $X_1, \cdots, X_K$.

C. Obtaining the Gaussian equivalent of the solution

Once the minimum of (7) is reached, it is desirable to obtain the covariance matrix Σ_K^a of the vector X_K^a, in order to connect the initialization step with the online approach of the next section.

The covariance matrix Σ_Φ of the optimized parameters Φ is easily obtained by using a local approximation of the term $Y - \hat{Y}$ in the vicinity of the minimum. The resulting Σ_Φ results from the following close expression:

$$\Sigma_\Phi = (J^T \Sigma_L J)^{-1}, \tag{11}$$

where J is the Jacobian matrix of $\hat{Y}$ with respect to parameters Φ. The Jacobian is available from the optimization method, in which is used to compute the iteration steps.

By truncating both the solution Φ and its covariance matrix Σ_Φ, the augmented vector X_K^a and its covariance matrix Σ_K^a are obtained.

V. Online Algorithm

In this section the solution to X_k^a given the last pose information is derived. The fact that last frame information is available and the assumption of soft motion between frames allows to greatly simplify the problem.

A special emphasis is given in this document to the fact that any process handled by the system is considered a random entity, in fact a Gaussian distribution defined at each case by its mean vector and covariance matrix. The problem of obtaining pose and structure, encoded in X_k^a given image observations Y_k and the last pose information X_{k-1}^a is viewed from the point of view of statistical inference, which means searching for the posterior probability distribution $p(X_k^a|Y_1, \cdots, Y_k)$. That distribution gives the best estimation of X_k^a given all the past knowledge available.

The online approach is divided into three steps:

- **Estimation Step:** using the previous pose X_{k-1}^a and the motion model a Gaussian distribution which infers the next state is given $p(X_k|Y_1, \cdots, Y_K)$.
- **Robust Layer:** the correspondence problem in this point easily fails, so for each camera a number of unlabeled outliers pollute the measurement vector Y_k. Using a robust algorithm and the information contained in the state vector, the outliers are discarded before the next step.
- **Correction Step:** using an outlier-free measurement vector, we are confident to use all the information available to obtain the target posterior distribution $p(X_k^a|Y_1, \cdots, Y_k)$

In all three steps we would manage the idea of propagating statistics over non-linear functions (f and h). We show how to face the problem using first order expansions as it offers more compactness and is more readable. However as is stated in [19] there are other methods for Gaussian

propagation (e.g. the Unscented Transformation) with better statistical performance and that are less biased than the first order expansion we show here.

A. Estimation Step

The estimation step uses the motion models available to infer the next pose of the robot. Such transit is often a process of uncertainty addition, as the motion information is not given accurately or only linear motion models are available. It includes an update of the mean and covariance of the last pose as follows:

$$X^a_{k|k-1} = g^a(X^a_k, U_k) \tag{12}$$

$$\Sigma^a_{k|k-1} = J_X^T \Sigma^a_{k-1} J_X + J_U^T \Sigma_W J_U, \tag{13}$$

where J_X and J_U are the first derivatives of the function g^a with respect to X^a_{k-1} and U_k respectively. Usually J_X in odometry systems is the identity, so at this step the covariance matrix $\Sigma^a_{k|k-1}$ results to be bigger in terms of eigenvalues, which means uncertainty.

It must be noticed that the motion model g^a leaves untouched the structure points contained in the state vector as we suppose that the object is rigid.

B. Correction Step

The correction step removes the added uncertainty in the estimation by using image measurements. It passes from the distribution $p(X^a_k|Y_1, \cdots, Y_{k-1})$ to the target distribution $p(X^a_k|Y_1, \cdots, Y_k)$, which includes the last measurement.

It is mandatory to remark that the measurement vector in the online process Y_k does not share the same structure that the one used in the initialization processes, where each camera observes a separated set of points. Instead, due to robot's motion, any point is suitable to be seen by any camera and a group of cameras can see the same point.

Using the estimation shown in (12), and knowing the correspondence between measurements with the camera and structure point of the state vector, the estimated measurement is given:

$$Y_{k|k-1} = h^a(X^a_k) \tag{14}$$

$$\Sigma_{Y_{k|k-1}} = J_h^T \Sigma^a_{k|k-1} J_h + \Sigma_V \tag{15}$$

$$\Sigma_{X^a Y} = \Sigma^a_{k|k-1} J_h, \tag{16}$$

where J_h is the Jacobian matrix of the function h^a with respect to X^a_k and Σ_V is block diagonal matrix with Σ_v on each block.

The correction step itself is a linear correction of $X^a_{k|k-1}$ and $\Sigma^a_{k|k-1}$ by means of the Kalman gain K_G:

$$K_G = \Sigma_{X^a Y} \Sigma^{-1}_{Y_{k|k-1}} \tag{17}$$

$$X^a_k = X^a_{k|k-1} + K_G(Y_k - Y_{k|k-1}) \tag{18}$$

$$\Sigma^a_k = \Sigma^a_{k|k-1} - K_G \Sigma^T_{X^a Y} \tag{19}$$

As it is stated in (19) the resulting Σ^a_k is reduced compared to $\Sigma^a_{k|k-1}$ which means that after the correction step, the uncertainty is “smaller”.

C. Robust Layer

The robust layer has the objective of removing bad measurements from Y_k to avoid inconsistent updates of X^a_k in the correction step. We propose and extend the same idea proposed in [19], in which a RANSAC algorithm [28] is used between the estimation and correction step. The general idea is to found among the measured data Y_k a set which is agree in the solution of X^a_k they will give at the correction step.

In the literature, the RANSAC algorithm uses a close form solution for the problem of inferring X_k^a given the measurements Y_k or alternatively a geometrical constraint that can be fitted such as fundamental matrixes or the trifocal tensors. In this paper we propose to avoid such complex structures and to take advantage of the versatility which offers the Kalman filter. We make no distinction of which measurement comes for which camera, and implement RANSAC sampling over the entire set of measurements Y_k. For each subset of Y_k used in RANSAC, the correction step is used to obtain the most voted X_k^a. This approach allows for example to include the information of all cameras at once without needing of camera grouping (e.g. combinations of two for fitting fundamental matrixes).

Some previous definitions are needed prior to use introduce RANSAC:

- ϵ is the expected probability of "outliers" inside measurements.
- From the measurement vector Y_k which has in general N_Y elements, groups of s measurements, denoted as Y_k^s are selected randomly. [1].
- It is defined as $X_k^a = F(X_{k|k-1}^a, Y_k^s)$ which computes the pose X_k^a using a minimum number s of measurements. This function is implemented using the correction step expressions shown in B reduced to use measures grouped inside Y_k^s.
- The function $d = D(X_k^a, y_k^i)$ gives the Mahalanobis distance d between a single measurement y_k^i taken from Y_k and its estimation $\hat{y}_k^i$ using X_k^a.

$$d = (y_k^i - \hat{y}_k^i)\Sigma_{y_k^i}^{-1}(y_k^i - \hat{y}_k^i)^T, \tag{20}$$

  where $\Sigma_{y_k^i}$ is obtained by propagating the statistical properties of X_k^a through the projection model.
- The probability of finding at least a set of s measurements Y^s free from "outliers" is defined as p (e.g $p = 0.99$).

[1] The upperindex s must not be misled with the notation we used in IV in which the upperindex was used to indicate the camera

- N_{in} is the number of correct correspondences associated to a model X_k
- N_{in}^{best} as the biggest number of inliers found in a set of correspondences associated to the best solution for X_{best}^a.
- The number of iterations N^{it}, necessary to achieve p has a close form:

$$N^{it} = \frac{log(1-p)}{log(1-(1-\epsilon)^s)} \tag{21}$$

Finally the general RANSAC method is presented in Algorithm 1:

Algorithm 1 RANSAC

1: $N^{it} = 0$, $N_{in}^{best} = 0$, X_{best}^a, $N = 1$, $p = 0.99$.
2: **while** $N^{it} < \min(N, N_{max}^{it})$ **do**
3: Random sampling of $y^1, \cdots y^N$ to obtain a set Y^s with s elements.
4: A model hypothesis is obtained $X = g_d(Y^s, X^a)$ and the set $Y^{in} = \{m_i \,|\, |g(X, y^i, M^i)| < d_{max}\}$ with size N_{in}.
5: **if** $N_{in} > N_{in}^{best}$ **then**
6: $X_{best}^a = X^a$, $\epsilon = N_{in}/N$.
7: $N^{it} = \frac{log(1-p)}{log(1-(1-\epsilon)^s)}$
8: **end if**
9: $N^{it} = N^{it} + 1$
10: **end while**

The number of measurements s used to randomly sample Y_k is defined to be $s = 4$ as this is the minimum required to compute the pose of a 3D object using image measurements.

VI. RESULTS

Our proposal is tested using synthetic generated data and real images taken in a room with four cameras.

A. *Synthetic data*

The synthetic data is generated artificially according to the conditions which will be encountered in a real configuration. A maximum of $N_c = 8$ cameras are situated approximately forming a rectangle of $2m \times 3m$ as it is shown in Fig. 4. The robot's geometry is formed by a set of points randomly distributed inside a cylindrical volume of half a meter of radius and one meter height. The robot follows a differential motion model over the ground plane, ruled by its angular ω_k and linear speed v_k encoded in $U_k = (\omega_k(^o/s), v_k(mm/s))^T$. The vector U_k is affected by a motion noise with covariance $\Sigma_w = diag(\sigma_v^2 = 1, \sigma_\omega^2 = 10)$ The pose of the robot is thus composed of the 2D position in the ground plane (x_k, y_k) and the single orientation angle θ_k. The measurement noise is fixed to $\Sigma_V = 10 \cdot I_{2\times 2}$. The intrinsic parameters of each camera are variations of those encountered in a low cost sensor with 640×480 pixels of resolution with a CCD sensor of $1/3''$ size and a optic with a focal length of $6mm$. The trajectory described by the robot consists of a circumference of radius $2m'$ which takes place inside the common area viewed by the cameras. Both the initialization and online experiments run using such trajectory, so that we can compare the accuracy in fair circumstance.

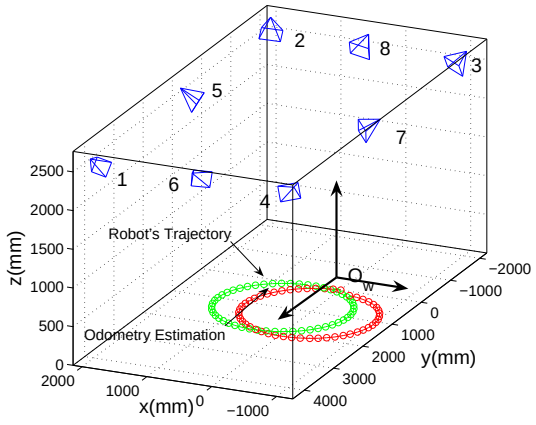


Fig. 4. Distribution of cameras and robot's trajectory used to generate synthetic data.

The experiments are divided on those dedicated to the initialization method proposed and those

used to test the online algorithm.

A.1 Initialization method

In this experiments we consider that each camera n observes a number $N_n = 6$ points from the robot's structure. The trajectory is downsampled, so that $K = 50$ positions. We observe that adding more time samples does not achieve better accuracy.

The following experiments are proposed:

- Error in pose (Fig. 6.a) and geometry (Fig. 6.c) versus % of outliers in the measurements.
- Error in pose (Fig. 6.b) and geometry (Fig. 6.d) versus % of outliers in the measurements, when the robust cost function is used.
- Error in pose (Fig. 5.e) and geometry versus (Fig. 6.f) % of the circular path made.

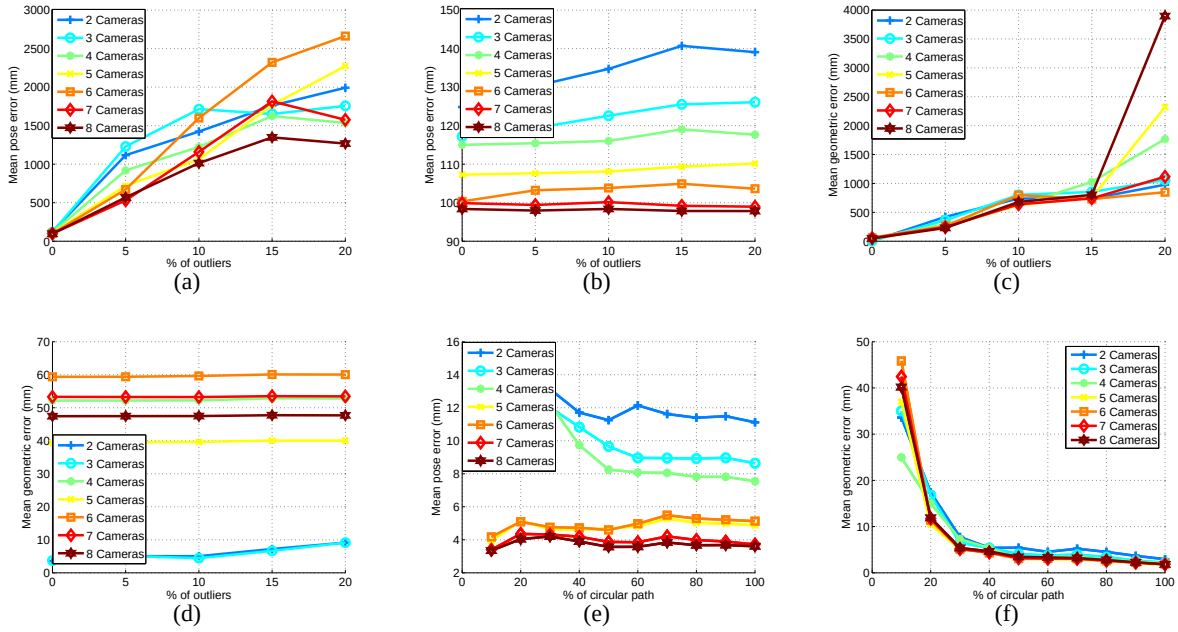


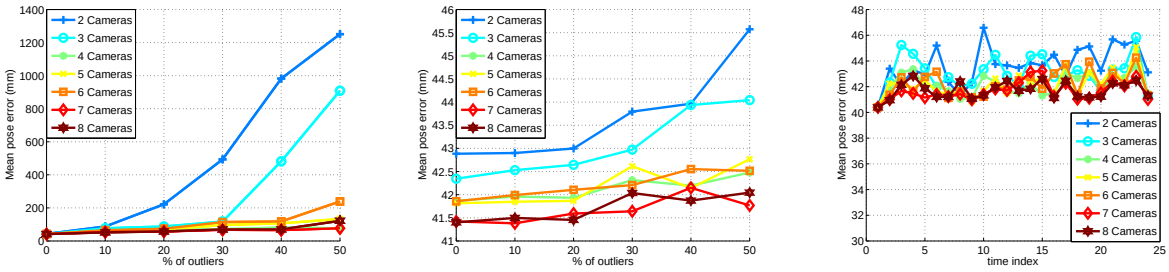
Fig. 5. Experiments of the initialization method using synthetic data.

A.2 Online method

The robust approach of the initialization algorithm is used to set up the pose and geometry X_K^a. The online algorithm covers the circular path starting were the online algorithm ended.

The following experiments are proposed:

- Error in pose and geometry versus % of outliers without robust layer. (Fig. 6.a)
- Error in pose and geometry versus % of outliers. (Fig. 6.b)
- Error in pose and geometry versus time. (Fig. 6.c)



(a) (b) (c)

Fig. 6. Experiments of the online method using synthetic data.

B. Real Results

The real experiment is compound of four cameras (see Fig. 7.c), filling the same area showed in the initialization, and a mobile robot which presents also the same kind of motion model used for the synthetic data. In Fig. 7.a, Fig. 7.b, Fig. 7.d and Fig. 7.e, the projection of the three-dimensional model (Fig. 7.d) obtained from the robot is shown on each of the four cameras.

In Fig. 8 the trajectory obtained using our proposal is shown, compared to the one obtained using odometry from the robot, which is highly noisy. Using a manual annotation procedure, it has been assessed that the estimation given by our proposal is not biased in the whole trajectory.

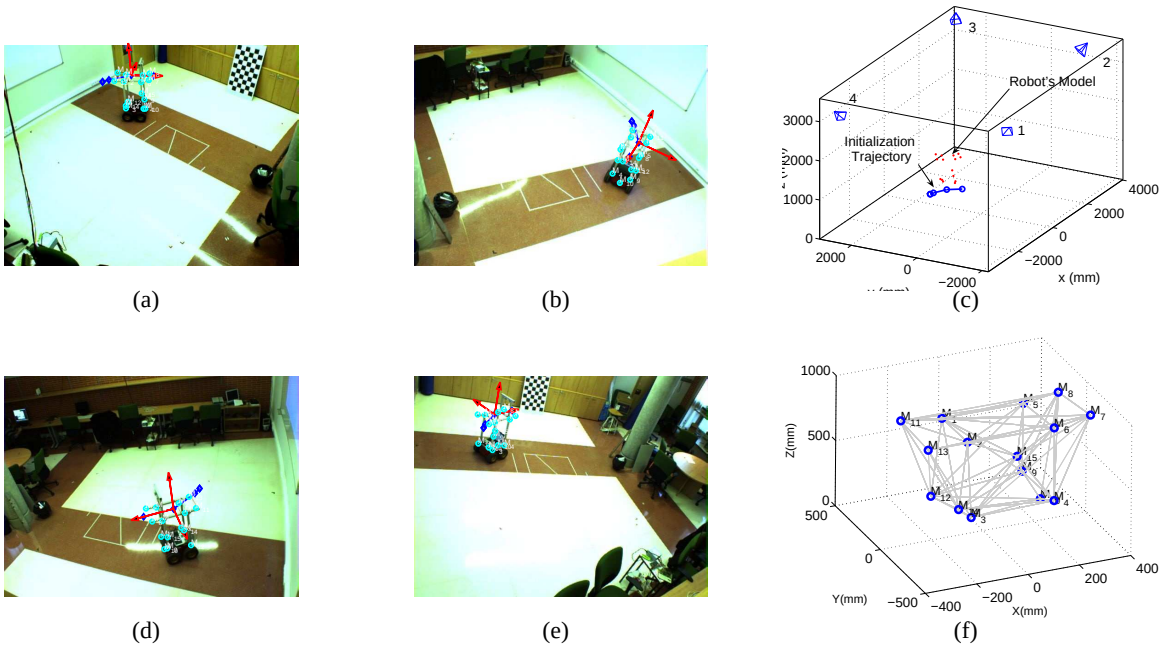


Fig. 7. Experiments of the initialization method using real data.

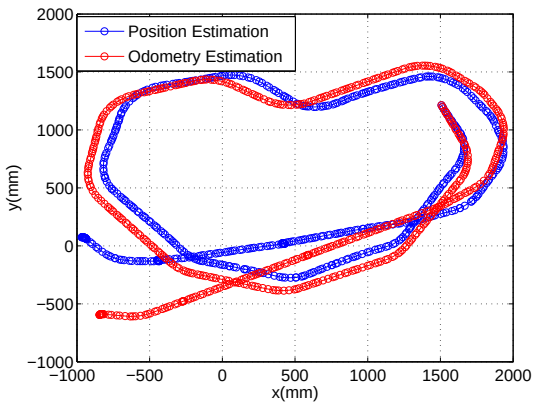


Fig. 8. Comparison between the online estimation and the odometry system.

VII. CONCLUSIONS

This paper has proposed a system that achieves robot localization using several cameras without needing invasive beaconing on the robot or supervised learning tasks. Compared to the single camera solution, proposed in [19], which this paper extends, the usage of several cameras allows to avoid the needed of using odometry systems in the robot in any stage of the algorithm, which reduces the required knowledge from the object to localize.

The two steps which compound the system, initialization of robot's pose and geometry, and the online process are designed so that they are robust against the inclusion of outliers in the algorithm, which is of importance to achieve a reliable solution.

The tests using synthetic data show that the more the cameras, the better in general becomes the algorithm in terms of accuracy in both geometry and pose estimation. The algorithm is tested using in real conditions, performing the localization of a robot using four cameras inside a room. Using visual inspection , the solution do not present bias in a long trajectory and behaves well even when the object is far from the cameras. Our proposal shows promising results as a reliable robot localization system, but also any other rigid object. The extension to tackle multiple robots is quite straightforward using our approach, as the robust layers allows to efficiently remove measurements that do not behave like individual rigid objects.

REFERENCES

[1] A. Pentland, "Smart rooms," *Scientific American*, 1996.

[2] H. Hashimoto, J.H. Lee, and N. Ando, "Self-identification of distributed intelligent networked device in intelligent space," *Robotics and Automation, 2003. Proceedings. ICRA'03. IEEE International Conference on*, vol. 3, 2003.

[3] J.H. Lee and H. Hashimoto, "Controlling mobile robots in distributed intelligent sensor network," *Industrial Electronics, IEEE Transactions on*, vol. 50, no. 5, pp. 890–902, 2003.

[4] P. M. Newman and J. J. Leonard, "Consistent convergent constant time SLAM," in *Proceedings of the International Joint Conference on Artificial Intelligence*, 2003.

[5] A. J. Davison, "SLAM with a single camera," in *Workshop on Concurrent Mapping and Localization for Autonomous Mobile Robots, in conjunction with ICRA*, 2002.

[6] Y. Hada, E. Hemeldan, K. Takase, and H. Gakuhari, "Trajectory tracking control of a non-holonomic mobile robot using igps and odometry," in *Multisensor Fusion and Integration for*

Intelligent Systems, MFI2003. Proceedings of IEEE International Conference on, 30 July-1 Aug. 2003, pp. 51–57.

[7] I. Fernández, M. Mazo, J.L. Lázaro, D. Pizarro, E. Santiso, P. Martín, and C. Losada, "Guidance of a mobile robot using an array of static cameras located in the environment," *Autonomous Robots*, vol. 23, no. 4, pp. 305–324, 2007.

[8] Kazuyuki Morioka, Xuchu Mao, and Hideki Hashimoto, "Global color model based object matching in the multi-camera environment," in *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, Oct. 2006, pp. 2644–2649.

[9] Jaeyong Chung, Namgyu Kim, Jounghyun Kim, and Chan-Mo Park, "Postrack: a low cost real-time motion tracking system for vr application," in *Virtual Systems and Multimedia, 2001. Proceedings. Seventh International Conference on*, Oct 2001, pp. 383 – 392.

[10] T.Sogo, H.Ishiguro, and T.Ishida, "Acquisition of qualitative spatial representation by visual observation," in *IJCAI99*, 1999.

[11] Eckhard Kruse and Friedrich M. Wahl, "Camera-based observation of obstacle motions to derive statistical data for mobile robot motion planning.," in *ICRA*, 1998, pp. 662–667.

[12] Adam Hoover and Bent David Olsen, "Sensor network perception for mobile robotics.," in *IEEE International Conference on Robotics and Automation ICRA'00*, 2000, pp. 342–347.

[13] Peter Steinhaus, Marcus Ehrenmann, and Rdiger Dillmann, "Mephisto. a modular and extensible path planning system using observation," *Lecture Notes in Computer Science*, vol. 1542, pp. 361–375, 1999.

[14] D. Nister, "Preemptive RANSAC for live structure and motion estimation," *Machine Vision and Applications*, vol. 16, no. 5, pp. 321–329, 2005.

[15] AW Fitzgibbon and A. Zisserman, "Automatic 3D model acquisition and generation of new images from video sequences," *Proceedings of European Signal Processing Conference*, pp.

1261–1269, 1998.

[16] F. Schaffalitzky and A. Zisserman, "Multi-view matching for unordered image sets, or How do I organize my holiday snaps?," *Proc. ECCV*, vol. 1, pp. 414–431, 2002.

[17] Iryna Gordon and David G. Lowe, *Toward Category-Level Object Recognition*, chapter What and where: 3D object recognition with accurate pose, pp. 67–82, Springer-Verlag, 2006.

[18] R. Szeliski and SB Kang, "Recovering 3D shape and motion from image streams using non-linearleast squares," *Computer Vision and Pattern Recognition, 1993. Proceedings CVPR'93., 1993 IEEE Computer Society Conference on*, pp. 752–753, 1993.

[19] Daniel Pizarro, Enrique Santiso, Manuel Mazo, and Marta Marron., "Pose and sparse structure of a mobile robot using an external camera," in *Proceedings of the IEEE International Symposium on Intelligent Signal Processing*, 2007, pp. 389–394.

[20] Jianbo Shi and Carlo Tomasi, "Good features to track," in *1994 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'94)*, 1994, pp. 593 – 600.

[21] David G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[22] H. Moravec, *Robot Rover Visual Navigation*, Ann Arbor, Michigan: UMI Research Press, 1981.

[23] C. Harris and M. Stephens, "A combined corner and edge detector," in *Fourth Alvey Vision Conference*, Manchester, 1988.

[24] David G. Lowe, "Object recognition from local scale-invariant features," in *International Conference on Computer Vision*, Greece, September 1999, pp. 1150–1157.

[25] V. Lepetit, L. Vacchetti, D. Thalmann, and P. Fua, "Fully automated and stable registration for augmented reality applications," *Mixed and Augmented Reality, 2003. Proceedings. The Second IEEE and ACM International Symposium on*, pp. 93–102, 2003.

[26] B.D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," *Proc. DARPA Image Understanding Workshop*, vol. 121, pp. 130, 1981.

[27] Bill Triggs, P. McLauchlan, Richard Hartley, and A. Fitzgibbon, "Bundle adjustment – a modern synthesis," in *Vision Algorithms: Theory and Practice*. 2000, vol. 1883, pp. 298–372, Springer-Verlag.

[28] R. C. Bolles. M. A. Fischler, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Comm. of the ACM*, vol. 24, pp. 381.395, 1981.