

Current state of the art of vision based SLAM

Naveed Muhammad, David Fofi, Samia Ainouz
Le2i UMR CNRS 5158, IUT Le Creusot, Université de Bourgogne, France

ABSTRACT

The ability of a robot to localise itself and simultaneously build a map of its environment (Simultaneous Localisation and Mapping or SLAM) is a fundamental characteristic required for autonomous operation of the robot. Vision Sensors are very attractive for application in SLAM because of their rich sensory output and cost effectiveness. Different issues are involved in the problem of vision based SLAM and many different approaches exist in order to solve these issues. This paper gives a classification of state-of-the-art vision based SLAM techniques in terms of (i) imaging systems used for performing SLAM which include single cameras, stereo pairs, multiple camera rigs and catadioptric sensors, (ii) features extracted from the environment in order to perform SLAM which include point features and line/edge features, (iii) initialisation of landmarks which can either be delayed or undelayed, (iv) SLAM techniques used which include Extended Kalman Filtering, Particle Filtering, biologically inspired techniques like RatSLAM, and other techniques like Local Bundle Adjustment, and (v) use of wheel odometry information. The paper also presents the implementation and analysis of stereo pair based EKF SLAM for synthetic data. Results prove the technique to work successfully in the presence of considerable amounts of sensor noise. We believe that state of the art presented in the paper can serve as a basis for future research in the area of vision based SLAM. It will permit further research in the area to be carried out in an efficient and application specific way.

Keywords: SLAM, vision based SLAM

1. INTRODUCTION

Simultaneous Localisation and Mapping (SLAM) is the problem of a robot being autonomously able to build a map of an unknown environment and simultaneously localising itself in the environment. This ability makes a robot truly autonomous.

A considerable amount of research has been carried out on SLAM using vision sensors during the last decade. Cameras provide rich information about the environment enabling the detection of stable features. Furthermore cameras are low-cost, light and compact, easily available, offer passive sensing and have low power consumption. All these features make cameras very attractive to be used for SLAM.

This article gives current state of the art on visual SLAM and is organised as follows. Section 2 of the article surveys different imaging systems used for visual SLAM, different types of features extracted from the environment, types of landmark initialisations for visual SLAM, different SLAM techniques, and use of wheels odometry for visual SLAM. Section 3 compares some approaches discussed in section 2, and includes a summarising table for state-of-the-art visual SLAM techniques. Section 4 presents the implementation of stereo-vision based EKF SLAM for a synthetic dataset. The article concludes in section 5.

2. CLASSIFICATION OF VISION BASED SLAM TECHNIQUES

This section surveys state-of-the-art vision based SLAM techniques. A classification of state-of-the-art vision based SLAM techniques is presented in terms of (i) imaging systems used for performing SLAM, (ii) features extracted from the environment, (iii) initialisation of landmarks, (iv) SLAM algorithms, and (v) use of wheel odometry information. Table 1 presents a summary of this classification.

Email: Naveed.Muhammad@laas.fr, David.Fofi@u-bourgogne.fr, Samia.Ainouz@insa-rouen.fr

2.1 Imaging systems

Different imaging systems have been used for visual SLAM including single cameras, stereo pairs, multiple camera rigs and catadioptric sensors. The pros and cons of each of these imaging systems have been discussed below with pointers to some of the published implementations.

2.1.1 Single camera

Single Camera SLAM is also referred to as Bearing-only SLAM as a single image provides only the direction of features present in robot's environment and does not provide the depth information. To get the 3D location of a feature multiple images from different viewpoints are required. Some visual SLAM implementations using single cameras are [1], [2], [3], [4], [5], [6]. Wide-angle cameras (above 90° field of view) have also been used for visual SLAM [7], [8], as they enable the tracking of features over wider motion ranges. Most of the single camera SLAM implementations mentioned above are for indoor environments. [2] and [9] show implementations of single camera visual SLAM in outdoor environment.

2.1.2 Stereo pair

A stereo pair can provide 3D location of the observed features in the environment, this makes a stereo pair readily usable for visual SLAM. [9] have implemented visual SLAM using stereo pairs for ground and aerial robots. [10] also shows an implementation of visual SLAM using stereo pair.

2.1.3 Multiple camera rigs

Multiple camera rigs have also been used for visual SLAM. One advantage is that the use of multiple cameras increase the field of view and enables the tracking of features over wider robot motion. Another advantage is that the spatial resolution over the field of view of a multiple camera rig is uniform unlike the catadioptric sensors which also offer a large field of view. [11] have used an eight camera rig which offers 360° field of view, to carry out visual SLAM.

2.1.4 Catadioptric sensors

Catadioptric sensors are attractive for application in visual SLAM because they offer a wide field of view. Catadioptric sensors have been used in different formations for visual SLAM. [12] shows implementation using a single catadioptric sensor mounted on a ground robot. [13] have used two catadioptric sensors as a stereo pair.

[14] use a catadioptric sensor consisting of a single camera and two fixed conic mirrors as shown in Figure 1. This type of sensor provides two views of the scene in a single image. The advantage of using this type of sensors (instead of two catadioptric sensors as a stereo pair) is that corresponding points in two views of the scene lie on radial lines in the image reducing the complexity of stereo matching process.



Figure 1. Single camera double mirror catadioptric sensor used by [14].

2.2 Features from environment

In order to carry out SLAM using vision, features from the environment have to be extracted that can be used as landmarks for SLAM. These features have to be stable and observable from different view points and angles. Many types of environment features have been used for SLAM using vision.

2.2.1 Point features

Point features are the most commonly used features for visual SLAM. Harris corner detector has widely been used in the recent years for interest point detection as in [2], [11], [12], [9]. [15] suggest that Harris corner detector is the most suitable interest point detector for visual SLAM.

To apply Harris corner detector to an image first the image gradients I_x , and I_y in x and y directions respectively are calculated for each pixel. Then a matrix C is calculated at each pixel using an image patch around the pixel:

$$C(x, y) = \begin{pmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_y I_x & \sum I_y^2 \end{pmatrix}$$

Let λ_1 and λ_2 be the Eigen values of above matrix C , an auto-correlation function R is defined as:

$$R = \lambda_1 \lambda_2 - k (\lambda_1 + \lambda_2)^2$$

where k is a constant. Sharply peaked values of R represent corners in the image under consideration. [16].

[1] use Harris-Laplace detector for the detection of interest points. Harris-Laplace detector uses a scale adapted Harris function to detect interest points [16]. The detection operator of Shi and Tomasi has been used for interest point detection in [7] and [8].

Scale-Invariant Feature Transform (SIFT) has also been used for interest point detection for visual SLAM as in [3], [14], [17]. SIFT features are invariant to scaling, translation and rotation and are partially invariant to illumination changes and 3D projection [18]. SIFT features are key locations in state space detected at maxima or minima of difference of Gaussian function [18].

As the robot moves, it has to (i) associate the newly observed features with the previously seen features and (ii) recognise new features in order to create new landmarks when required. This data association is a crucial stage in SLAM using any type of sensors because wrong data association can lead to extremely erroneous localisation and mapping. Once the interest points have been detected using an interest point detector, the data association can be performed using different methods. [7] and [8] use normalised cross-correlation between planar patches around interest points for data association. [10] and [12] use local groups of interest points to perform robust data association.

2.2.2 Line/edge features

Line/Edge features exist in abundance in structured environments. This type of features is more useful for mapping than point features in that they provide some geometrical information about the environment unlike point features. Moreover these features are invariant to lighting [6] and significant viewpoint changes [5]. [5] use Plücker coordinates to represent line features in the environment and use them for carrying out SLAM. [6] use Canny's detector to extract edges from the images and use portions of these edges as features which they name *edgelets*. The advantage of using edgelets over using complete edges is that the complete edges can at times be partially occluded or broken into multiple edges in the image [6].

2.2.3 Featureless approaches

SLAM using vision has also been carried out without explicitly extracting any features from the environment. [4] have presented a technique in which they take 640x480 image, convert it into greyscale, take a 300x160 pixels sub-window at

centre of the image and compute an *image array* by summing the pixel values in each column of the sub-window. By comparing the shift in image arrays of consecutive frames, the rotation and speed of the moving camera is extracted as follows:

Rotation $\Delta\theta$:

$$f(s) = \frac{1}{w - |s|} \sum_{n=1}^{w-|s|} |I_{n+\max(s,0)}^{k+1} - I_{n-\min(s,0)}^k|$$

$$\Delta\theta = \alpha(\arg \min f(s))$$

where I represents the image array values for images k and $k+1$, w is the image width, and s is the shift between two image arrays.

Speed v :

$$v = \frac{1}{w - |s_m|} \sum_{n=1}^{w-|s_m|} |I_{n+\max(s_m,0)}^{k+1} - I_{n-\min(s_m,0)}^k|$$

where s_m is the image array shift corresponding to the best rotation match. [4].

2.3 Initialisation of landmarks

To carry out SLAM, some of the extracted features from environment are used as landmarks. In order to be used as landmarks, both direction and depth information of the feature has to be estimated. In general two types of landmark initialisation approaches exist.

2.3.1 Undelayed approaches

To get direction and depth information of a feature right at initialisation of the SLAM system, one way is to use a stereo pair. [9] show an implementation of SLAM using stereo vision for ground and aerial robots. Another way of getting an undelayed initialisation of landmarks is to use an artificial target of known appearance for the SLAM system to initialise. [7] use a solid rectangle printed on a paper as initialisation target for the SLAM system. The featureless approach of [4] explained in sub-section 2.2.3 is also a special case in the category of undelayed approaches.

2.3.2 Delayed approaches

When a single camera is used to carry out SLAM without the aid of any artificial target at start-up, determining the depth of features detected in the first acquired frame is not possible, using the first frame alone. In this case the camera is moved to slightly different view points and corresponding features are matched in different frames. This enables estimating the depth of some features which are then used as landmarks to initialise the SLAM system. [1], [2] and [3] give implementations of delayed landmark initialisation using multiple frames from single camera.

2.4 SLAM techniques

Different SLAM techniques exist and have been implemented using visual sensors. These include the well known *Extended Kalman Filtering*, *Particle Filtering* and some other techniques.

2.4.1 Extended Kalman Filtering (EKF)

As the robot performs SLAM, at a time instant k , let x_k be the vector representing the current robot state (position and orientation), u_k be the control input applied at time $k-1$ to move the robot to state x_k (this can also be the wheel odometry readings during the interval $(k-1, k]$). Let m be the set representing the locations of all landmarks and z_k be the set of landmark observations at time instant k .

In EKF, the motion model (model that gives the robot state x_k from state x_{k-1} and control input u_k) is described as:

$$x_k = f(x_{k-1}, u_k) + w_k$$

where function f models the robot kinematics and w_k accounts for the un-modelled kinematics and noise and is considered to be zero mean uncorrelated Gaussian noise with covariance Q_k . Similarly the observation model (the model for observation of landmarks m from robot at state x_k) is described as:

$$z_k = h(x_k, m) + v_k$$

where function h describes the observation geometry and v_k accounts for the observation errors and is zero mean uncorrelated Gaussian noise with covariance R_k . [19].

EKF is performed in two steps, first the robot motion is predicted using control input and then updated using the landmarks observation. For both the steps the sets representing the mean of robot state (x_k) and landmark locations (m_k) is calculated and a matrix P is calculated representing covariances within and between robot state and landmark locations, and can be represented as:

$$P_{k|k} = \begin{bmatrix} P_{xx} & P_{xm} \\ P_{xm} & P_{mm} \end{bmatrix}_{k|k}$$

At time instant k the prediction step is performed as follows:

$$\bar{x}_{k|k-1} = f(\bar{x}_{k-1|k-1}, u_k)$$

$$\text{and } P_{xx, k|k-1} = \nabla f P_{xx, k-1|k-1} \nabla f^T + Q_k$$

where ∇f represents the Jacobian of f calculated at the estimate $\bar{x}_{k-1|k-1}$. The update step is performed as follows:

$$\begin{bmatrix} \bar{x}_{k|k} \\ \bar{m}_k \end{bmatrix} = \begin{bmatrix} \bar{x}_{k|k-1} \\ \bar{m}_{k-1} \end{bmatrix} + W_k [z_k - h(\bar{x}_{k|k-1}, \bar{m}_{k-1})]$$

$$\text{and } P_{k|k} = P_{k|k-1} - W_k S_k W_k^T$$

where:

$$S_k = \nabla h P_{k|k-1} \nabla h^T + R_k$$

$$W_k = P_{k|k-1} \nabla h^T S_k^{-1}$$

∇h is the Jacobian of h calculated at $\bar{x}_{k|k-1}$ and \bar{m}_{k-1} . [19].

$[z_k - h(\bar{x}_{k|k-1}, \bar{m}_{k-1})]$ is called the *innovation* and it represents the difference between observation and prediction, S_k is the *innovation covariance* and W_k is the Kalman Gain.

By far EKF is the most used SLAM technique. One problem with EKF is that their computational cost grows quadratically with the number of landmarks. Secondly they use linearised models of non-linear motion and observation models [19]. Some implementations of EKF for visual SLAM can be found in [1], [7], [5], [8], [12], [9] and [13].

2.4.2 Particle Filtering

Particle Filters have also been successfully employed in visual SLAM. The FastSLAM algorithm introduced by [20] used particle filter to estimate robot pose and EKF for estimating landmark locations.

The FastSLAM algorithm is based on the fact that in SLAM problem, if robot's pose is known, the individual landmark measurements are independent. In other words if the robot poses are known, the estimation of landmark locations can be decoupled into independent estimation problems for each of the landmarks. [20].

This leads to factorisation of the combined SLAM state as follows:

$$P(X_{0:k}, m|Z_{0:k}, U_{0:k}, x_0) = P(m|X_{0:k}, Z_{0:k}) P(X_{0:k}|Z_{0:k}, U_{0:k}, x_0)$$

where $X_{0:k}$, $Z_{0:k}$ and $U_{0:k}$ represent the sets of all robot states, observations and control inputs respectively from time 0 to k . [19].

At the time instant k , the joint distribution is represented by the set:

$$\{w_k^{(i)}, X_{0:k}^{(i)}, P(m|X_{0:k}^{(i)}, Z_{0:k})\}_i^N$$

where N is the total number of particles, $w_k^{(i)}$ is the importance weight given to the i th particle and

$$P(m|X_{0:k}^{(i)}, Z_{0:k}) = \prod_j^M P(m_j|X_{0:k}^{(i)}, Z_{0:k})$$

where M is the total number of landmarks. [19].

For robot state at time instant k , each particle calculates the new robot state using the state at time $k-1$ and control input u_k . This creates a temporary set of particles. This temporary set is sampled by giving different weights to different particle and this gives the final set of particles representing the robot state at time instant k . For each particle, each landmark is updated using a separate Kalman filter. [20].

[3], [6] and [17] perform visual SLAM using techniques based on particle filtering.

2.4.3 RatSLAM

RatSLAM is a SLAM technique based on the model of rodent hippocampus. Rodents have *place fields* which are patterns of neural activity that correspond to locations in space, and are modulated by rodent motion and visual sensing. This technique uses a competitive attractor network as the approximation of the rodent hippocampus. Activity packets in the attractor network represent pose hypotheses. The attractor network is called *pose cells*. Wheel odometry information is used to inject activity in pose cells and thus shifting the activity packets, this is the process of “path integration”. Visual sensing information is converted into *local view* representation. If the current visual scene is familiar, it also injects activity into the pose cells that are linked to current scene. [21]. The RatSLAM structure is shown in Figure 2.

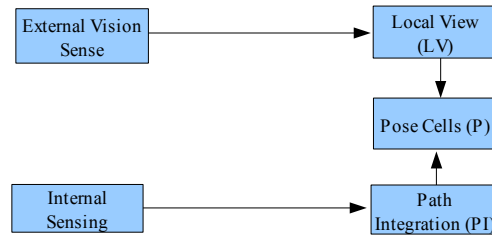


Figure 2: RatSLAM structure by [21].

A pose cell unit excites the units close to it and inhibits the units that are far from it. This process leads to the dominance of an activity packet. When activity is injected close to the dominating activity packet, the activity tends to move the

packet towards itself, whereas the activity injected far from the dominating packet creates a new packet which competes with the dominating packet and can also eventually become the dominating packet. [21].

During the path integration process, when the robot translates, the pose cell activity is shifted in x,y plane and magnitude of this shift depends on translational velocity of the robot. Similarly if the robot rotates, the activity is shifted in θ direction and the magnitude of shift depends on the rotational velocity of the robot. [21].

The visual sensing information is encoded in the *local view cells*. Association between the active local view cells and highly active pose cells is strengthened by changing the weight of connections between them. This is done using Hebbian learning, and is expressed as follows:

$$\beta_{(ijk)(lmn)}^{t+1} = \beta_{(ijk)(lmn)}^t + \eta (P_{lmn} V_{ijk})$$

where β is the strength of connection between local view cell and pose cell, η is the learning rate, V and P are the activation levels of the local view and pose cells respectively, and ijk and lmn represent the spaces in which local view and pose cells are represented respectively. [21].

This SLAM technique that has been used in the featureless implementation of [4].

2.4.4 Local bundle adjustment

SLAM is similar to the problem of *Structure from Motion (SFM)* where the movement of a camera and the positions of the observed points are estimated. Generally SFM is performed off-line by computationally expensive global bundle adjustment optimisation, and because of high computation time these algorithms are not feasible for real-time applications in SLAM. [2] have proposed a method which uses *fast and local* bundle adjustment in order to carry out SLAM in real-time using a single camera.

When the SLAM system initialises, three acquired frames are used to set up the global coordinate system. The system uses Harris corners as interest points and the points are matched between frames by computing Zero Normalised Cross Correlation in the regions of interest. As new frames are acquired during the SLAM process, some frames are selected as *key-frames*. In order to understand the camera pose estimation process, consider at a point in time when we have already calculated the camera poses C_1 to C_{i-1} which correspond to the key-frames I_1 to I_{i-1} . A set of points and projections of these points in the corresponding images are also known. Now a new frame I is acquired and we have to estimate the corresponding camera pose C . Frame I is matched with last key-frame I_{i-1} to find a set of points whose projections on the cameras C_{i-2} , C_{i-1} and C are known and whose 3D coordinates have already been computed earlier. Now the pose C is estimated using a pose estimation algorithm and RANSAC which is then refined using a fast LM optimization. [2].

As the new frames are acquired, if the number of matching points between the current frame and last key-frame is less than a threshold, preceding frame (the frame acquired before the current frame) is selected as a new key-frame. Similarly when the uncertainty of estimated position is very high, a new key-frame is added to the system. As the new key-frame I_i is added, new points which are observed only in the current three key-frames are reconstructed using triangulation. At the addition of the new key-frame, a local bundle adjustment is carried out by Levenberg-Marquardt minimisation of the cost function $f^i(C^i, P^i)$ where C^i and P^i are the camera poses and 3D points selected for the current optimisation stage. For this stage n last cameras and N last frames are used where N is greater than or equal to n . C^i represents the set $\{C_{i-n+1}, \dots, C_i\}$ and P^i is the set of all 3D points projected on cameras in set C^i . The function f^i is given by:

$$f^i(C^i, P^i) = \sum_{C_i \in \{C_{i-n+1}, \dots, C_i\}} \sum_{P_j \in P^i} (\varepsilon_{ij}^2)$$

with $\varepsilon_{ij}^2 = d^2(p_{ij}, K_i p_j)$

where ε_{ij}^2 is the squared Euclidean distance between the estimated projection of point p_j on camera C_i and the corresponding measured position. K_i is the i th projection matrix consisting of corresponding extrinsic and intrinsic parameters. [2].

[2] have found that $n = 3$ or 4 and $6 \leq N \leq 11$ are sufficient values for the above local bundle adjustment.

2.5 Use of wheel odometry

Classically SLAM is carried out using both the wheels odometry information and data from other sensors sensing the environment. Process uncertainties account for inaccuracies in odometry data and other noise. [7] suggest that even if no wheel odometry is available, the whole camera motion can be modelled as process uncertainty or noise. Many visual SLAM implementations do not use the odometry data to carry out SLAM including [2], [4], [6], [7], [8] and [17], whereas many implementations use wheel odometry along with other sensors to carry out SLAM as in [1], [3], [5] and [11].

3. DISCUSSION

3.1 Monocular and stereo imaging systems

[9] have experimented and compared monocular and stereo approaches to visual SLAM for ground and aerial robots. Figure 3 taken from [9] shows the evolution of robot pose for stereo and monocular approaches with camera(s) mounted frontwards and sidewards on a ground robot that takes three loops on a 6m diameter circular trajectory.

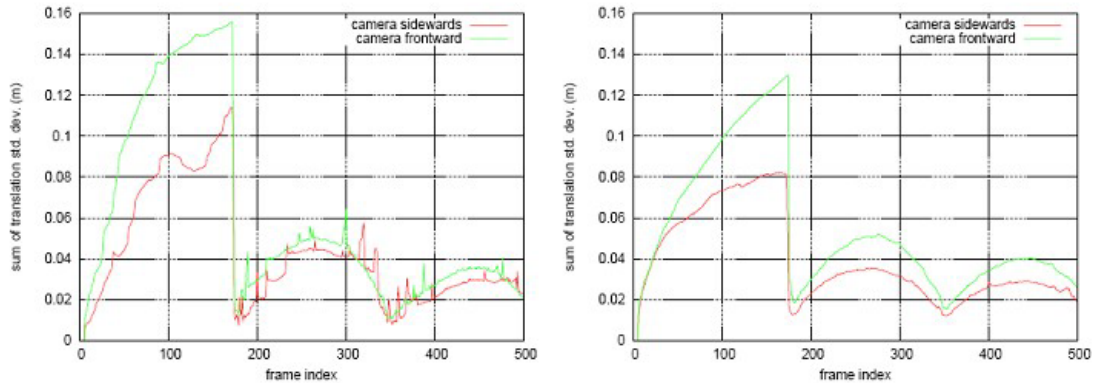


Figure 3. Evolution of robot pose uncertainty for (left) stereo and (right) monocular approaches, by [9].

It can be observed that the uncertainty significantly drops at the first loop closure. It can also be observed that the uncertainties are lower for cameras mounted sidewards. This is because of the fact that in sidewards case the features are tracked on more frames and in monocular case the base-line between the two consecutive frames is greater which results in fast initialisation of landmarks. Here monocular SLAM approach seems to slightly outperform the stereo approach in terms of uncertainty in robot pose. This is because in case of stereo, the features are to be matched between the two images from the stereo pair and also between the two consecutive acquisitions in time. This results in fewer matches in some frames in the stereo case, whereas the matching problem is relatively less complicated in the monocular case. [9].

In terms of consistency of robot pose estimate, [9] suggest that bearing-only SLAM (monocular approach) does not perform as good as stereo SLAM.

3.2 Interest point detectors

[16] have studied the suitability of different interest point detectors for the application in visual SLAM. They have experientially compared the robustness of five interest point detectors i.e. Harris, Harris-Laplace, SUSAN (Smallest Univalued Segment Assimilating Nucleus), SIFT (Scale Invariant Feature Transform) and SURF (Speeded Up Robust Features), against changes in viewpoint and scale. In their experiment, [16] acquired 12 image sequences each containing

21 images with a viewpoint change of 2.5° between every two consecutive frames; and to study the robustness against change in scale they acquired 14 image sequences each containing 12 images where camera moved 0.1 m between every two consecutive acquisitions.

Figure 4 (left) shows the repeatability of the features detected in the first frame, in the proceeding frames as the viewpoint changes gradually. Harris corner detector outperforms other interest points detectors as it is able to maintain 30% of the initially detected features till the last frame with 50° change in viewpoint. Similarly Figure 4 (right) shows repeatability of features with change in scale. In this case, Harris corner detector also outperforms other interest point detectors.

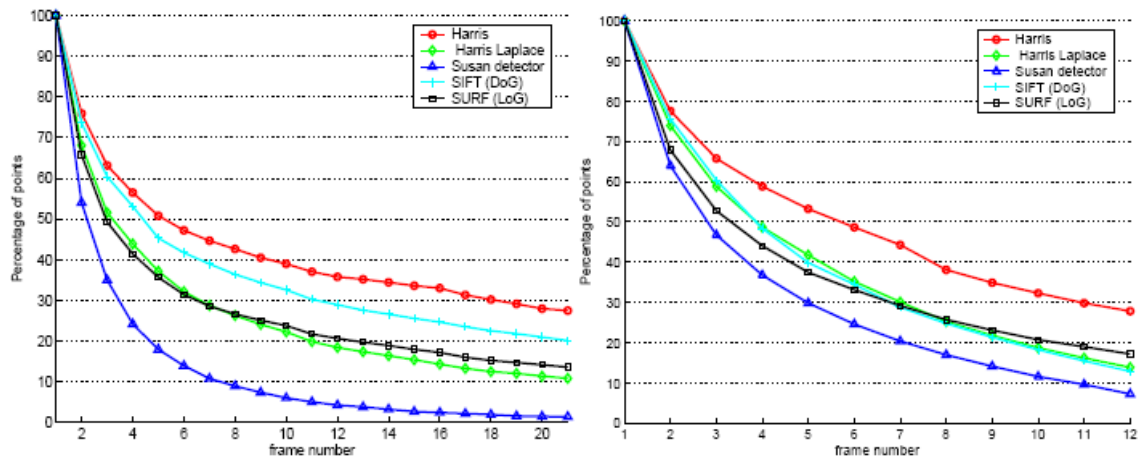


Figure 4. Repeatability of detected features with changes in viewpoint (left) and scale (right), by [16].

3.3 Summarising table

Table 1 gives a classification of state-of-the-art visual SLAM techniques.

4. EKF VISION BASED SLAM IMPLEMENTATION

4.1 Synthetic stereo-pair dataset

A synthetic dataset was developed in order to permit the implementation of EKF visual SLAM. The dataset consisted of stereo-pair images taken along a trajectory as the stereo camera pair moved in a synthetic structured environment. The Epipolar Geometry Toolbox developed for Matlab by [22] provided a framework for creation and visualisation of 3D environment and development of the dataset. Figure 5 shows the 3D environment and a pair of images taken at a single time instant by the stereo pair.

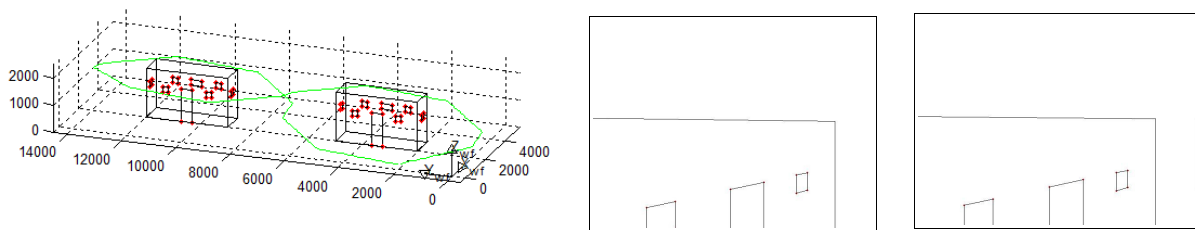


Figure 5. (Left) Synthetic 3D environment and ground truth robot trajectory, (right) a stereo image pair.

4.2 EKF SLAM

Extended Kalman Filtering based SLAM was implemented on the developed stereo-pair dataset. Considerable amounts of noise were added to the information from sensors including robot position, robot velocity and landmark measuring (vision) sensors. Noise in robot position measurement was considered higher than noise in robot velocity and landmark position measurements. Figure 6 shows the robot trajectory estimated using EKF SLAM.

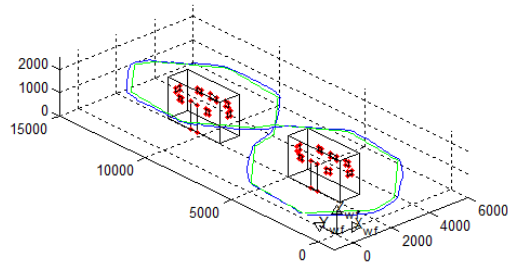


Figure 6. Ground truth (green/light) and estimated (blue/dark) robot trajectories (units in mm).

Results show that stereo-pair based EKF SLAM performs well in a synthetic 3D environment. Table 2 shows overall mean absolute errors in the x , y and z coordinates of robot position and velocity estimates. Results show that the error in x and y coordinates of both position and velocity estimates are greater than the errors in z coordinates. This is because of the fact that robot moved smoothly in z direction compared to its motion in x and y directions.

Results show that stereo-pair based EKF SLAM performs well in a synthetic 3D environment. Table 2 shows overall mean absolute errors in the x , y and z coordinates of robot position and velocity estimates. Results show that the error in x and y coordinates of both position and velocity estimates are greater than the errors in z coordinates. This is because of the fact that robot moved smoothly in z direction compared to its motion in x and y directions.

Table 2. Overall mean absolute estimation errors (units in mm and mm/time stamp)

Pos x	Pos y	Pos z	Vel x	Vel y	Vel z
110.6	102.7	18.7	21.8	23.3	3.8

More information on the development of synthetic dataset and EKF SLAM implementation described in this section can be found in [23].

5. CONCLUSION

Simultaneous Localisation and Mapping (SLAM) is an ability that is necessary for mobile robots to be able to autonomously move and perform the required tasks in an unknown environment. Vision sensors are attractive to be employed for SLAM because of a number of reasons including their rich sensing, easy availability and cost effectiveness. This article has presented some state-of-the-art visual SLAM techniques. Implementation of Extended Kalman Filtering based visual SLAM on a synthetic dataset proved the technique to be efficient and successful even in the presence of significant noise in robot position, velocity, and landmark position sensing. We believe that vision sensors have a huge potential to be employed for SLAM, and gradually more robust visual SLAM methods are being developed.

ACKNOWLEDGMENTS

The work was carried out during first author's "Masters in Vision and Robotics (VIBOT)" Course, which was funded by Europeans Commission's "Erasmus Mundus Program".

REFERENCES

- [1] Jensfelt, P., Kragic, D., Folkesson, J. and Björkman, M., "A Framework for Vision Based Bearing Only 3D SLAM", International Conference on Robotics and Automation, Orlando, FL, 2006.
- [2] Mouragnon, E., Lhuillier, M., Dhome, M., Dekeyser, F. and Sayd, P., "Monocular Vision Based SLAM for Mobile Robots", International Conference on Pattern Recognition, Hong Kong, 2006.

- [3] Goncalves, L., Di Bernardo, E., Benson, D., Svedman, M., Ostrowski, J., Karlsson, N. and Pirjanian, P., “A visual Front-end for Simultaneous Localization and Mapping”, International Conference on Robotics and Automation, Barcelona, Spain, 2005.
- [4] Milford, M. and Wyeth, G., “Featureless Vehicle-Based Visual SLAM with a Consumer Camera”, Australasian Conference on Robotics and Automation, Brisbane, Australia, 2007.
- [5] Lemaire, T. and Lacroix, S., “Monocular-vision based SLAM using line segments”, International Conference on Robotics and Automation, Roma, Italy, 2007.
- [6] Eade, E. and Drummond, T., “Edge Landmarks in Monocular SLAM”, British Machine Vision Conference, Edinburgh, UK, 2006.
- [7] Davison, A.J., Reid, I.D., Molton, N.D. and Stasse, O., “MonoSLAM: Real-Time Single Camera SLAM”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 29, no. 6, 2007, pp 1052-1067.
- [8] Davison, A.J., Cid, Y.G. and Kita, N., “Real-Time 3D SLAM with Wide-Angle Vision”, IFAC/EURON Symposium on Intelligent Autonomous Vehicles, Lisbon, Portugal, 2004.
- [9] Lemaire, T., Berger, C., Jung, I. and Lacroix, S., “Vision-Based SLAM: Stereo and Monocular Approaches”, International Journal of Computer Vision, vol. 74, no. 3, 2007, pp 343-364.
- [10] Berger, C. and Lacroix, S., “Using Planar Facets for Stereovision SLAM”, HAL – CCSD e-articles, available at <http://hal.archives-ouvertes.fr/hal-00174889/en/>, 2007.
- [11] Kaess, M. and Dellaert, F., “Visual SLAM with a Multi-Camera Rig”, Technical Report GIT-GVU-06-06, Georgia Institute of Technology, 2006.
- [12] Lemaire, T. and Lacroix, S., “SLAM with panoramic vision”, Journal of Field Robotics, vol. 24, no. 1-2, 2007, pp. 91-111.
- [13] Kim, J. and Chung, M.J., “SLAM with Omni-directional Stereo Vision Sensor”, International Conference on Intelligent Robots and Systems, Las Vegas, Nevada, 2003.
- [14] Kim, J., Yoon, K., Kim, J. and Kweon, I., “Visual SLAM by Single-Camera Catadioptric Stereo”, SICE-ICASE International Joint Conference, Busan, Korea, 2006.
- [15] Ballesta, M., Gil, A., Mozos, Ó.M., and Reinoso, Ó., “Local Descriptors for Visual SLAM”, Workshop on Robotics and Mathematics, Coimbra, Portugal, 2007.
- [16] Mozos, Ó.M., Gil, A., Ballesta, M., and Reinoso, Ó., “Interest Point Detectors for Visual SLAM”, Conference of the Spanish Association for Artificial Intelligence, Salamanca, Spain, 2007.
- [17] Elinas, P., Sim, R. and Little, J.J., “ σ -SLAM: Stereo Vision SLAM Using the Rao-Blackwellised Particle Filter and a Novel Mixture Proposal Distribution”, International Conference on Robotics and Automation, Orlando, FL, 2006.
- [18] Lowe, D.G., “Object Recognition from Local Scale-Invariant Features”, International Conference on Computer Vision, Corfu, Greece, 1999.
- [19] Durrant-Whyte, H. and Bailey, T., “Simultaneous Localization and Mapping: Part I”, IEEE Robotics & Automation Magazine, 2006, pp 99-108.
- [20] Montemerlo, M., Thrun, S., Koller, D. and Wegbreit, B., “FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Problem”, National Conference on Artificial Intelligence, Edmonton, Canada, 2002.
- [21] Milford, M. J., Wyeth, G.F. and Prasser, D., “RatSLAM: A Hippocampal Model for Simultaneous Localization and Mapping”, International Conference on Robotics and Automation, New Orleans, LA, 2004.
- [22] Mariottini, G. L. and Parattichizzo, D., “EGT for Multiple View Geometry and Visual Servoing”, IEEE Robotics & Automation Magazine, 2005, pp 26-39.
- [23] Muhammad, N., “Vision Based Simultaneous Localisation and Mapping for Mobile Robots”, Masters Thesis, Universite de Bourgogne, France, 2008.

