Simuntaneous Localisation and Mapping with a Single Camera

Abhishek Aneja and Zhichao Chen

3 December, 2004

Simuntaneous Localisation and Mapping with a Single Camera

1 Abstract

Image reconstruction is common in the field of computer vision these days. But, reconstructing the image as well as at the same time tracking the camera position is a field still being worked on. This task of estimating camera motion from measurements of a continuously expanding set of selfmapped visual features is one of a class of problems known as Simultaneous Localisation and Mapping (SLAM) in the robotics community. For a camera moving in an unknown scenes is a challenging problem when working in a real time rather than offline processing condition

We present a method for a single camera moving in a 2D and mapping of a sparse set of features using motion modelling and an information guided active measurement strategy. Where the location of the features and camrea motion is controlled by a simulator.

2 Introduction

Robot mapping has been an active area of research for the past few decades. It gives a spatial model of the physical environment through robot navigation. Structure from motion research in computer vision has reached the point where fully automated reconstruction of the trajectory of a video camera moving through a previously unknown scene is becoming routine [**?**],but these and other successful approaches seen to date have been formulated as off-line algorithms and required batch, simultaneous processing of all the images acquired in the sequence[2].Building a truly autonomous mobile robot, that would give the mapping of the unknown environment in real time, however, enforces hard constraints on the processing permissible and hence, still poses a great challenge.

Is there a need for autonomous mobile robots in real world ? They operate well in a defined region and need to be able to identify the limits of that region and avoid other potential hazards while carrying out its task. It may be necessary for a robot to monitor its position accurately and execute defnite movements, but possible to facilitate this in a simple way with external help such as a prior map with landmark beacons in known positions. Exploring remote or dangerous areas, although assistance may often still be possible here, require a mobile robot which can navigate truly autonomously [1].

Among previous work, that of Chiuso et al.[3] present a real-time, full-covariance Kalman Filter-based approach to sequential structure from motion, but aim towards model generation rather than localisation. Davison[2],successful obtained results, in real time, for a camera moving in a 3D world and correctly estimating the landmarks position. Feature visibility is calculated based on the relative position of the camera and feature, and saved position of the camera from which the feature was initialised. We present a method to track a robot moving in a 2D world and at the same time localise the feature points using "Extended Kalman Filter". The code has still not been tried for a real time sequence. It works well for a simulator created motion of the camera and features.

3 SLAM with First Order Uncertainity Propagation

Extended Kalman Filter (EKF)-based algorithms, propagating first-order uncertainty in the coupled estimates of robot and map feature positions, combined with various techniques for reducing computational complexity in large maps, have shown great success in enabling robots to estimate their locations accurately and robustly [2]. The overall state $\mathbf{x}$ of the system is represented by a vector which can be partitioned into the state $\mathbf{a}$ of the camera (or robot) and the state $\mathbf{p^{(i)}}$ of the enteries in the map surroundings. The state vector is accompanied by a covariance matrix $\mathbf{P}$,which represents the uncertainity, to first order in all quantities of the state vector.

$$\mathbf{x} = \begin{pmatrix} \mathbf{a} \\ \mathbf{p^{(1)}} \\ \mathbf{p^{(2)}} \\ \mathbf{p^{(3)}} \\ . \end{pmatrix} \mathbf{P} = \begin{pmatrix} P_{aa} & P_{ay1} & P_{ay2} & . & . \\ P_{y1a} & P_{y1y1} & P_{y1y2} & . & . \\ P_{y2a} & P_{xy1} & P_{xy2} & . & . \\ P_{y2a} & P_{y1y1} & P_{y2y2} & . & . \\ . & . & . & . & . \end{pmatrix}$$

Features estimates $\mathbf{p^{(i)}}$ can be added or deleted freely from the map as required, making the state vector $\mathbf{x}$ and $\mathbf{P}$ grow and shrink accordingly. In general, $\mathbf{x}$ and $\mathbf{P}$ vary in two steps: first, during motion, using **motion model** to calculate how the camera (or robot) moves and how its uncertainity increases. Second, when feature measurement are obtained, **measurement model** explains how the map and robot uncertainity can be reduced. The correlation between the position estimates of the cluster of close features whch are inherent in map building are uncertain with the world reference frame but highly correlated among themselves (i.e. their relative position are well known).Holding correlation information means that measurements of any one of this cluster correctly affects estimate of the others, and is the key to being able to re-visit and recognise known areas after periods of neglect.

4 Motion Model for a Smoothly Moving Camera

We define the world coordinate frame **World** to be fixed in world, and Camera coordinate frame **Camera** fixed with reapect to camera. The position vector $\mathbf{a}$ is made up of the location of the camera defined by $[a_x(t)a_y(t)a_\theta(t)]^{\mathbf{T}}$. We asume the camera to be moving

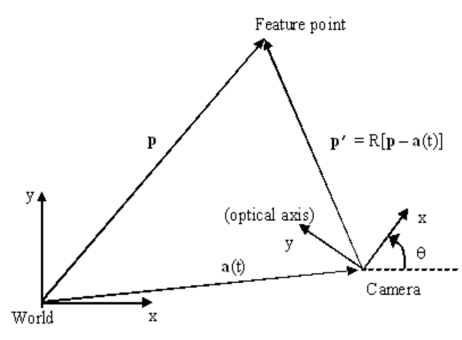


Figure 1: Frames and Vector Geometry

at a constant velocity,but statistical model of its motion in a time step is that on average undetermined accelerations occur with a Gaussian profle. So, we also need to add the velocity in the position vector **a**,which now becomes$a = [a_x(t)a_y(t)a_\theta(t)\dot{a}_x(t)\dot{a}_y(t)\dot{a}_\theta(t)]^T$where $\dot{a}_x(t), \dot{a}_y(t)$ represent a velocity in x and y direction and $\dot{a}_\theta(t)$ represent the angular velocity.

5 Visual Feature Measurements

In real sequence, as done by Davison[2], features were found using Shi and Tomasi Tracker[5]. In our experiments, we place the features randomly in the image and start off with some known features and estimate rest to be away from the orignal features.

In this section we consider the **measurement model** of the features already in the SLAM map. The estimate of the camera position **a** and feature position $\mathbf{p}^{(i)}$ allow the value of the measurement to be predicted. From figure 1, the relative position of a feature point to the camera is expected to be:

$$\tilde{\mathbf{p}}^{(i)}(t) = R[\mathbf{p}^{(i)} - \mathbf{a}(t)] \tag{1}$$

$$R(t) = \begin{bmatrix} \cos a_\theta & \sin a_\theta \\ -\sin a_\theta & \cos a_\theta \end{bmatrix}$$

where,$\tilde{\mathbf{p}}^{(i)}(t)$is the feature point relative to camera coordiante R is the Rotation Matrix

Now the projection of features on the image plane,i.e. Changing from camera cordinate frame to worls coordinate frame can be represented by:

$$\begin{pmatrix} \beta u \\ \beta \end{pmatrix} = \begin{bmatrix} kf & u_0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \cos a_\theta & \sin a_\theta \\ -\sin a_\theta & \cos a_\theta \end{bmatrix} \begin{pmatrix} p_x^{(i)} - a_x(t) \\ p_y^{(i)} - a_y(t) \end{pmatrix}$$

where u_0 is the principal point of the camera (in pixels). Putting these equations together yields

$$u^{(i)}(t) = u_0 + f\frac{(p_x^{(i)} - a_x(t))\cos(a_\theta) + (p_y^{(i)} - a_y(t))\sin(a_\theta)}{(p_y^{(i)} - a_y(t))\cos(a_\theta) - (p_x^{(i)} - a_x(t))\sin(a_\theta)} \tag{2}$$

The uncertainity in this predicton, represented in the covariance matrix P, gives the shape of the Gaussian pdf over image coordinates and choosing a number of standard deviations deffines an elliptical search window within which the feature should lie with high probability.

6 Experiments and Results

6.1 Simulation -1:Camera moving in a straight line and passing through the features

The velocity is assumed to be $1m/s$ in x direction and $2m/s$ in y direction. The initial position of the camera frame is assumed to be zero and due to the uncertainty in the initial measurements the covariance. For the unknown fratures the initial estimates starte at 10 m away from the orignal location of the feature both in x and y direction. The known features have zero uncertainity, so all the rows and columns in the covariance matrix P, corrosponding to these fratures have a value equal to zero. The uncertainity for the unknown features is high, so the $P_{xx}, P_{yy}, P_{xy}, P_{yx}$ are set to a high value (between 750 to 1000). The features were successfully estimated with the uncertainity as a low as 1.6 cm.

The results shown are for a $30step$ simulator with velocity of $1ms^{-1}$ and $2ms^{-1}$ in x and y direction. with 5 known features and 10 unknown features.

6.2 Simulation 2:Camera moves along a straight line with a limited degree visual cone

We start with 4 feature points known for the initialization of EKF. A total of 20 landmarks were mapped. The field of view of camera is 1200. The camera moves with constant velocity 0.05m/second alone the x axis. The rotation velocity of camera is 0. We set high prior covariance P0 for unknown feature point. At each step, a predictive update was performed based on constant velocity motion model, and then a measurement based on only one feature point per step. When a landmark is out of view, it will be removed from state matrix. The ellipses around these landmarks illustrate the residual uncertainty that remains after mapping, as specified by the covariance matrix. In order to compare the uncertainty, the directions of all the ellipses are set to uniform.

6.3 Simulation 3: Camera makes a circle.

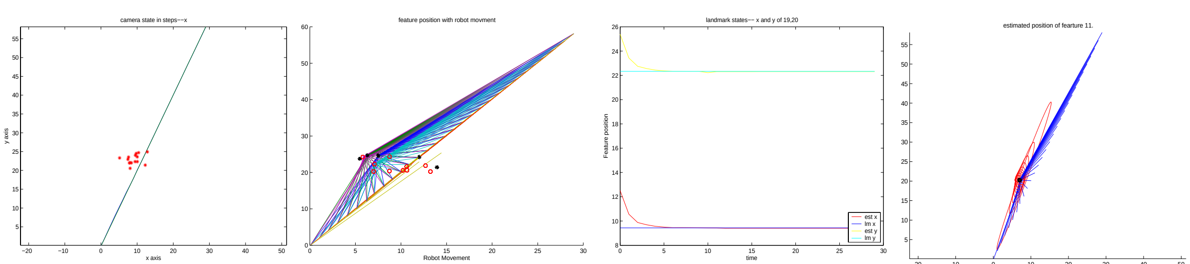
We still start with 4 feature points known for the initialization of EKF. A total of 20 landmarks were mapped, which are always in the field of the view of the camera—1200. The maximum velocity of translation is 0.1m/second and maximum velocity rotation is 40 /second. To make a circle, the camera states (velocity and rotation) have to vary continuously.

7 Conclusion

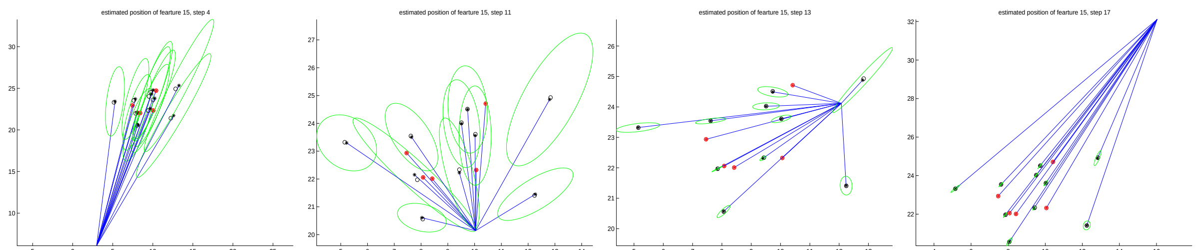
Simulation in 2D show the EKF with a single camera measurement model can identify the camera localization as well as landmark positions. With more measurements over time, the uncertainty ellipses shrink gradually. The development in this paper can also be extended for the 3D reconstruction.

Simulation shows if the known feature points are always in the field of view of camera, the estimates are more accurate and the system works stably. However, in the situation that the feature points are routinely in and out of the filed of view of camera, the estimate errors might be bigger and even result in failure of system. For the future research, we should focus on coping with accumulated error and increasing the stability of system

Results of First Expetiment

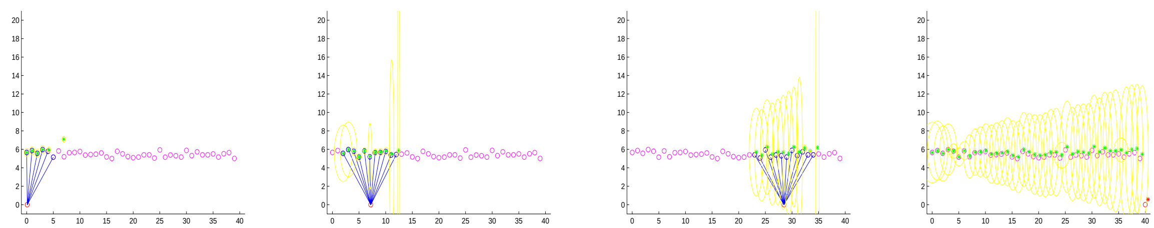


Camera and Features | Features and Camera | Feature over time | Uncertainity



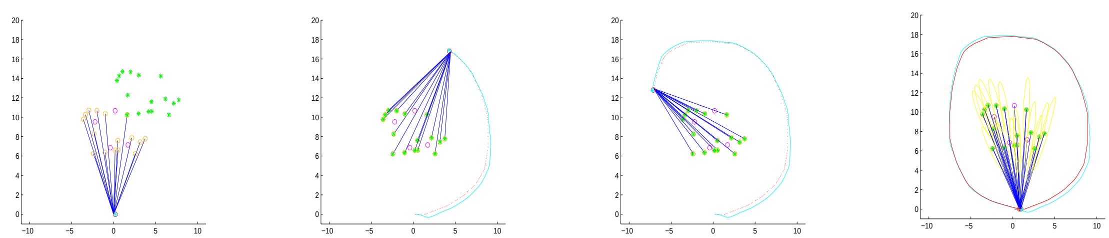
Step -04 Step 11 Step 13 Step 17

Results of Second Expetiment



Step 01 Step 70 Step 290 Final Step

Results of Third Expetiment



Step 01 Step 150 Step 250 Final Result

References

[1] Andrew Davison, D.Phil (Ph.D) Thesis, Mobile Robot Navigation Using ActiveVision , University of Oxford, 1998.

[2] Andrew J. Davison, Real-Time Simultaneous Localisation and Mapping with a Single Camera ,Proc. of the Internation Conference on Computer Vision, Nice, October 13 -16, 2003.

[3] A. Chiuso, P. Favaro, H. Jin, and S. Soatto. “MFm.”: 3-D motion from 2-D motion causally integrated over time. In Proceedings of the 6th European Conference on Computer Vision, Dublin, 2000.

[4] E. Foxlin. Generalized architecture for simultaneous localization, auto-calibration and map-building. In Proceedings of the IEEE/RSJ Conference on Intelligent Robots and Systems, 2002.

[5] J. Shi and C. Tomasi. Good features to track. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 593.600, 1994.