Chapter 1

REVIEW ON SENSORS AND RELATED WORK

1.1 SENSORS FOR OBSTACLE DETECTION

1.1.1 Overview

Many work in obstacle detection uses range sensors such as radar, laser range finder, LIDAR, sonar. [reference goes here]. Radar and laser range finder provides only point measurement at a given position and orientation. To acquire a full 3D range map of a scene, mechanical scanning mechanisms are required, which limits the data acquisition rate of these device. LIDAR operate in the same manner as laser range finder, except with the scanning mechanism built in. These sensors usually have high power requirement and mass, and may not be suitable for small and mid size UAV. Sonar is usually used in indoor or under water applications, and have wide beam profile which make it difficult to identify the origin of return signal, and results in low resolution range map. 3D flash LIDAR is capable of acquire 3D range measurement simultaneously by illuminating the entire field of view of the camera with a single laser, and capturing the reflected laser with a 2D imaging sensor [reference; wikipedia].

However, its high cost has limited its use in commercial application.

In recent years, many researches use optical sensor as a passive range sensor for its low weight, low cost. With the help of computer vision technology, optical sensors have been successfully used for range mapping and obstacle detection in a number of platforms [references]. There are several types of configuration in using optical sensor for range mapping: monocular, binocular, or multi-camera. Since optical sensors are bearing only sensors, the principle of range measurement is through triangulation a common scene point in two or more images captured. For binocular camera setups, two cameras are placed apart from each other with their relative geometry known and captures images simultaneously. If the position of a scene point can be accurately found in the images by both cameras, its distance can be calculated by using the difference in position of the projected point in images, and the separation of the cameras.

- Radar, sonar, laser range finder, 3D flash lidarvs. optical sensors
  - Radar, laser range finder have high power requirement and mass
  - Depth measurement can be obtained through optical sensors, which are inexpensive and light weight
  - Depth maps of a 3-D scene can be computed from a single pair of stereo camera. Stereo processing can require significant computational effort
- Monocular camera characteristic:
  - bearing-only sensor, which provide the measurement on the direction of the feature, and not the range. Other sensors, such as radar, are range and bearing sensors.

1.1.2 Monocular Vision and Binocular Vision

- Optical flow vs. feature detection and tracking
- The correspondence problem
- Initialization problem (addressed by Inverse depth parameterization)
- Lack of scale information of overall map -> must work with other sensors which provide robot motion measurement.

1.1.3 Limitation of Optical Sensor in Recursive Algorithm

- Error Accumulation over Iterations
  - Feature quality Decreases over Iterations

1.1.4 GPS and IMU

GPS and IMU are generally available on UAVs. These sensors provide a measurement on the robot motion. Odometry can provide the scale information which is missing in the bearing only measurement. Furthermore, odometry provides some prior information on the robot motion which can help to disambiguate the solution.

1.2 SLAM as A Sensor Fusion Framework

An essential aspect of autonomy for a mobile robot is the capability to determine its location. This capability is known as localization. Localization is typically a prerequisite for accomplishing real tasks, whether it is exploration, navigation toward a known goal, transportation of material, construction or site preparation. In many applications, the mobile robot has an a priori map. Given a map, the robot may localize by matching current sensor observations to features in the map. Given enough features and an unambiguous geometry, the pose of the robot can be determined or at least narrowed down to a set of possible locations.

Usable maps do not always exist, and it is not always possible to have accurate externally referenced pose estimates. If an a priori map is not available, the robot may need to construct one. With a precise, externally referenced position estimate from GPS or similar means, the robot can take its sensor observations, reference the observations to its current pose, and insert features in the map in the appropriate places. Without maps or externally referenced pose information, the robot must produce its own map and concurrently localize within that map. This problem has been referred to as concurrent localization and mapping (CLM) and simultaneous localization and mapping (SLAM).

1.2.1 Recursive Probabilistic Estimation using Extended Kalman Filter

The Kalman filter [1]published by R. E. Kalman in 1960 is a very powerful recursive data processing algorithm for dynamic stochastic processes. The filter find extensive use in control and navigation application for its ability of estimating past, present and even future state. It is an attractive candidate for data fusion framework as it can process all available measurements including previous knowledge of the process, regardless of their precision, to estimate the current value of the variable of interest. Given a dynamic process that satisfy the assumptions that Kalman filter is based on, the filter is the optimal algorithm in minimzing the mean of squared error of the state variable. This section briefly summerized assumption and formation of Kalman filter that's described in detail in [2] [3] [4] [5] [6]. A more intuitive introduction can be found in chapter 1 of [7].

The Kalman filter has three assumptions. 1) The system model is linear. The linearity is deisred in that the system model is more esily manipulated with engineering tool. When nonlinearities do exist, the typical approach is to linearize system model at some nominal points. 2)The noise embedded in system control and measurement is white. This assumption implies that the noise value is not correlated in time, and has equal power in all frequency. 3)The probability density function (PDF) of system and measurement noise is Gaussian. A Gaussian distribution is fully represented by the first and second order statistic (mean and variance) of a process. Most other densities require endless number of orders of statistic to describe the shape fully. Hence, when the probability density function of a noise process is non-Gaussian, the Kalman filter that propagates the first and second order statistic only include some of the information of the PDF, instead of all, as would be the case with Gaussian noise.

Kalman Filter Models

The Kalman filter requires two models. The process model define a discrete-time controlled process by a linear stochastic difference equation. The $n \times n$ matrix A relates the state variables x_{k-1} in previous time step $k-1$ to the state variabl x_k in the current time step k. The matrix B relates the optional control input μ to the state variables x. Given measurement vecor z_k of size $1 \times m$, the measurement model relates the state variables to the measurements by matrix H of size $n \times m$. The random variable w and v represent the uncertainty or noise of the process model, and measurement. w and v are assumed to be unrelated to each other, and has Gaussian distribution with covariance Q and R.

$$\text{Process Model: } x_k = Ax_{k-1} + B\mu_{k-1} + w_{k-1} \tag{1}$$

$$\text{Measurement Model: } z_k = Hx_k + v_k \tag{2}$$

The Algorithm

The Kalman filter operates in prediction and correction cycle after being initialized 1, with the state vector estimate $\hat{x}_k^-, \hat{x}_k^+$ contains the variable of interest at time step k, and state covariance matrix P_k^-, P_k^+ representing the error covariance of the estimate. The superscript - indicate the estimate is a priori (or predicted) estimate, and + indicate the estimate is a posteriori (or corrected) estimate. The equations used for prediction and correction are listed in 1. In prediction, an estimate of the state variables are made based on the known knowledge of the process (the process model). Since there are always unknown factor not fully described by the process model, the error of the estimate almost always increase in the prediction. During correction, a series of calculation were carried out to correct the a priori estimate. First, the predicted measurement $H\hat{x}_k^-$ are compared to the new measurement z_k. Their difference $z_k - H\hat{x}_k^-$ is called the measurement innovation, or residual. Next, the amount of residual is weighted by the Kalman gain K, and added to $\hat{x}_k^-$ as correction. The Kalman gain is fomulated so that it minimize the a posteriori error covariance matrix P_k^+.

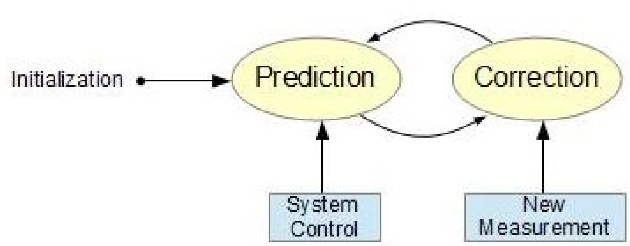

Figure 1: Kalman Operation Flow Diagram

Table 1: Kalman Filter Operation Equations

Prediction	$\hat{x}_k^- = A\hat{x}_{k-1}^+ + B\mu_{k-1}$	(3)
	$P_k^- = AP_{k-1}^+A^T + Q$	(4)
Correction	$K_k = P_k^- H^T(HP_k^- H^T + R)^{-1}$	(5)
	$\hat{x}_k^+ = \hat{x}_k^- + K_k(z_k - H\hat{x}_k^-)$	(6)
	$P_k^+ = (I - K_k H)P_k^-$	(7)

Extended Kalman Filter

For a discrete-time controlled process, or its relationship with the measurements are non-linear, a Kalman filter must be linearized about the estimated trajectory, and is referred to as an extended Kalman filter or EKF. A process with state vector x and measurement z that is governed by non-linear stochastic difference equation has process and measurement model

$$x_k = f(x_{k-1}, u_{k-1} + w_{k-1}), \tag{8}$$

$$z_k = h(x_k + v_k), \tag{9}$$

where the random varialbe w_k and v_k represent the process and measurement noise with variance Q and R. The the Kalman filter operation equations are given in table 2,

Table 2: Extended Kalman Filter Operation Equations

Prediction	$\hat{x}_k^- = f(\hat{x}_{k-1}^+, \mu_{k-1}, 0)$	(10)
	$P_k^- = A_k P_{k-1}^+ A_k^T + W_k Q_{k-1} W_k^T$	(11)
Correction	$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + V_k R_k V_k^T)^{-1}$	(12)
	$\hat{x}_k^+ = \hat{x}_k^- + K_k(z_k - h(\hat{x}_k^-, 0))$	(13)
	$P_k^+ = (I - K_k H) P_k^-$	(14)

where (subscript k omitted)

- A is the Jacobian matrix of partial derivatives of f with respect to x,

$$A_{[i,j]} = \frac{\partial f_i(\hat{x}_{k-1}^+, \mu_{k-1}, 0)}{\partial x_j}$$

- W is the Jacobian matrix of partial derivatives of f with respect to w,

$$W_{[i,j]} = \frac{\partial f_i(\hat{x}_{k-1}^+, \mu_{k-1}, 0)}{\partial w_j}$$

- H is the Jacobian matrix of partial derivatives of h with respect to x,

$$H_{[i,j]} = \frac{\partial h_i(\hat{x}_k^-, 0)}{\partial x_j}$$

- V is the Jacobian matrix of partial derivatives of h with respect to v,

$$V_{[i,j]} = \frac{\partial h_i(\hat{x}_k^-, 0)}{\partial v_j}$$

Note that when w and v directly describe the noise of state vector and measurement, the table 2 is the same as table 1.

Tuning

The tuning of the filter can be achieved by adjusting noise covariance matrix Q and R. The measurement noise covariance R is usually easy to estimate, ie., by taking some offline measurements to compute the variance. On the other hand, process noise covariance Q is more difficult. One common way is to inject enough uncertainty into the process noise, and only rely on reliable measurements.

1.2.2 SLAM with Extended Kalman Filter

The simultaneous localization and mapping(SLAM) problem answers to whether a robot can be make truly autonomous if it is plaed at an unknown location in an unknown environment. AFter many decades of research, the stucture of SLAM problem now has evolved to a standard Bayesian form. Two key approaches to solving a SLAM problem is to use the extended Kalman filter (EKF-SLAM) and through the use of Rao-Blackwellized particle filter (FastSLAM). A thourough tutorial to the SLAM problem is given in [8] [9].

General EKF Model for SLAM

There is a high degree of correlation betwee the estimates of the location of the landmarks in map [10] [11]. This correlation exists because of the common error in the estimated vehicle location [12]. The implication of these works is that a consistent full solution to the SLAM problem require a joint state composed of the vehicle pose and every landmark position [8]. When the vehicle pose and landmarks position are fomulated as one single estimation problem, the result was convergent. The correlation between landmark play an important role in the quality of the solution.

The more these correlation grew, the better the solution. [13] [14] [15] [16]

At a given time step k, the following variables are defined

- x_k describe the vehicle position and orientation.
- u_k the control vector applied at time $k-1$ to drive the vehicle to x_k at time k
- p_i a vector desecribe the location of the ith landmark. All landmarks locations are assumed to be time invariant.
- z_{ik} observation of the location of the ith landmark taken from the vehicle's location at time k

Then a complete state vector contains

$$\begin{bmatrix} \hat{x}_k \\ \hat{p}_k \end{bmatrix}$$

and the state covariance matrix contains

$$\begin{bmatrix} P_{xx} & P_{xp} \\ P_{px} & P_{pp} \end{bmatrix}$$

The prediction and correction procedure can then be carried out following the standard EKF formulation. The lamdmarks are generally not updated at the prediction (ie. only x_k and P_{xx} are updated) unless they are moving.

Properties of SLAM

Dissanayake [16] reached three important convergency properties of Kalman filter based SLAM, namely that:

1. *the determinant of any submatrix of the map covariance matrix decreases monotonically as observations are successively made;*

2. *in the limit as the number of observations increases, the landmark estimates become fully correlated;*

3. *in the limit the covariance associated with any single landmark location estimate reaches a lower bound determined only by the initial covariance in the vehicle location estimate at the time of the first sighting of the first landmark.*

These result shows that the complete knowledge of cross correlation between landmark estimates is critical in maintaining the structure of a SLAM problem. The error in estimates of landmarks becomes more and more correlated as the vehicle venture throuth the unknown environment. The error eventually becomes fully correlated which means given the location of one landmark, the location of any other landmark can be determined with absolute certainty. As the map converges, the error in the estimates of landmarks reduce to a lower bound determined by the error when the first observation was made.

The results above only refer to the evolution of covariance matrices computed by linear model. In reality, SLAM problem is nonlinear, and the computed covariance does not match the true estimation error. This leads SLAM consistency issue.

Linearization Error and Consistency

Many research report filter divergence due to linearization error. Literature review here: As defined in [17], a state estimator is consistent if the estimation errors (i) are zero-mean, and (ii) have covariance matrix smaller or equal to the one calculated by the filter.

Huang investigated [18] on properties and consistency of nonlinear two-dimentional EKF based SLAM problem, and conclude:

- Most of the convergence properties in [16] are still true for the nonlinear case provided that the Jacobians used in the EKF equations are evaluated at the true states.
- The main reasons for inconsistency in EKF SLAM are due to (i) the violation of some fundamental constraints governing the relationship between various Jacobians when they are evaluated at the current state estimate, and (ii) the use of relative location information from robot to landmarks to update the absolute robot and landmark location estimates.

The robot orientation uncertainty plays an important role in both the EKF SLAM convergence and the possible inconsistency. In the limit, the inconsistency of EKF SLAM may cause the variance of the robot orientation estimate to be incorrectly reduced to zero.

Linearization error can be interpreted as error resulted from calculating the Jacobian at the estimated state (wrong state) instead of the true state.

Camera Centric Coordinate System

SLAM for Large Scale Maps

Appendix A

Coordinate Transformation

For a mobile robot traveling in world. A point in space has coordinate $\begin{bmatrix} x & y & z \end{bmatrix}$ in world frame, and $\begin{bmatrix} x' & y' & z' \end{bmatrix}$ in the mobile frame. The two coordinate is related by

$$\begin{matrix} x \\ y \\ z \\ 1 \end{matrix}_{World} = Q \cdot \begin{matrix} x' \\ y' \\ z' \\ 1 \end{matrix}_{mobile} \tag{15}$$

where Q is the transformation matrix composed by rotation matrix on X, Y, and Z axis and a translation matrix. Q follows the TR?Y convention. . . (fig)

$Q(r_x, r_y, r_z, T) = Q_{Rz}(r_z) \cdot Q_{Ry}(r_y) \cdot Q_{Rx}(r_x) \cdot Q_T(T)$

$Q^{-1}(r_x, r_y, r_z, T) = Q_T^{-1}(T) \cdot Q_{Rx}^{-1}(r_x) \cdot Q_{Ry}^{-1}(r_y) \cdot Q_{Rz}^{-1}(r_z)$

$$Q_{Rx}(r_x) = \left[\begin{array}{cccc} 1 & 0 & 0 & 0 \\ 0 & \cos(r_x) & -\sin(r_x) & 0 \\ 0 & \sin(r_x) & \cos(r_x) & 0 \\ 0 & 0 & 0 & 1 \end{array}\right]$$

$$Q_{Ry}(r_y) = \left[\begin{array}{cccc} \cos(r_y) & 0 & \sin(r_y) & 0 \\ 0 & 1 & 0 & 0 \\ -sin(r_y) & 0 & \cos(r_y) & 0 \\ 0 & 0 & 0 & 1 \end{array}\right]$$

$$Q_{Rz}(r_z) = \left[\begin{array}{cccc} \cos(r_z) & -\sin(r_z) & 0 & 0 \\ \sin(r_z) & \cos(r_z) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{array}\right]$$

$$Q_T = \left[\begin{array}{cccc} 1 & 0 & 0 & T_x \\ 0 & 1 & 0 & T_y \\ 0 & 0 & 1 & T_z \\ 0 & 0 & 0 & 1 \end{array}\right]$$

$$Q_{Rx}^{-1} = Q_{Rx}^T$$

$$Q_{Ry}^{-1} = Q_{Ry}^T$$

$$Q_{Rz}^{-1} = Q_{Rz}^{T}$$

$$Q_T^{-1} = Q_T?$$

List of References

[1] R. E. Kalman. "A new approach to linear filtering and prediction problems." *Journal of Basic Engineering* **82**(1), 35–45. ISSN 0098-2202 (1960).

[2] H. Sorenson. "Least-squares estimation: from gauss to kalman." *IEEE Spectrum* **7**(7), 63–68. ISSN 0018-9235 (1970).

[3] Analytic Sciences Corporation. *Applied optimal estimation*. M.I.T. Press, Cambridge, Mass. ISBN 0262200279 (1974).

[4] M. S. Grewal. *Kalman filtering: theory and practice*. Prentice-Hall information and system sciences series. Prentice-Hall, Englewood Cliffs, N.J. ISBN 013211335X (1993).

[5] F. L. Lewis. *Optimal estimation: with an introduction to stochastic control theory*. Wiley, New York. ISBN 0471837415 (1986).

[6] R. G. Brown. *Introduction to random signals and applied Kalman filtering*. John Wiley & Sons, Inc., New York, 2nd ed edition. ISBN 0471525731 (1993).

[7] P. S. Maybeck. *Stochastic models, estimation and control*. Number v.141 in Mathematics in science and engineering. Academic Press, New York. ISBN 0124807011 (1979).

[8] H. Durrant-Whyte and T. Bailey. "Simultaneous localization and mapping: part i." *IEEE Robotics Automation Magazine* **13**(2), 99–110. ISSN 1070-9932 (2006).

[9] T. Bailey and H. Durrant-Whyte. "Simultaneous localization and mapping (SLAM): part II." *IEEE Robotics Automation Magazine* **13**(3), 108–117. ISSN 1070-9932 (2006).

[10] R. C. Smith and P. Cheeseman. "On the representation and estimation of spatial uncertainty." *The International Journal of Robotics Research* **5**(4), 56–68. ISSN 0278-3649, 1741-3176 (1986).

[11] H. Durrant-Whyte. "Uncertain geometry in robotics." *IEEE Journal of Robotics and Automation* **4**(1), 23–31. ISSN 0882-4967 (1988).

[12] J. Leonard and H. Durrant-Whyte. "Simultaneous map building and localization for an autonomous mobile robot." In "IEEE/RSJ International Workshop on Intelligent Robots and Systems '91. 'Intelligence for Mechanical Systems, Proceedings IROS '91," pages 1442–1447 vol.3 (1991).

[13] H. Durrant-Whyte, D. Rye, and E. Nebot. "Localization of autonomous guided vehicles." In G. Giralt and G. H. P. Dr.-Ing, editors, "Robotics Research," pages 613–625. Springer London. ISBN 978-1-4471-1257-0, 978-1-4471-1021-7 (1996).

[14] M. Csorba, J. K. Uhlmann, and H. F. Durrant-Whyte. "New approach to simultaneous localization and dynamic map building." volume 2738, pages 26–36 (1996).

[15] M. Csorba. *Simultaneous Localisation and Map Building* (1997).

[16] M. W. M. G. Dissanayake, P. Newman, S. Clark, H. Durrant-Whyte, and M. Csorba. "A solution to the simultaneous localization and map building (SLAM) problem." *IEEE Transactions on Robotics and Automation* **17**(3), 229–241. ISSN 1042-296X (2001).

[17] G. P. Huang, A. Mourikis, and S. Roumeliotis. "Analysis and improvement of the consistency of extended kalman filter based SLAM." In "IEEE International Conference on Robotics and Automation, 2008. ICRA 2008," pages 473–479 (2008).

[18] S. Huang and G. Dissanayake. "Convergence and consistency analysis for extended kalman filter based SLAM." *IEEE Transactions on Robotics* **23**(5), 1036–1049. ISSN 1552-3098 (2007).