**Kalman Filter**

* Kalman filter is the Bayes filter for the gaussian Linear case
* Performs recursive state estimation
* *Prediction* step to exploit the controls
* *Correction* step to exploit the observations

Bayes filter is one tool for state estimation

* Prediction
* Correction

Kalman Filter

* KF is a Bayes filter
* Everything is gaussian
* Optimum solution for linear models and gaussian distributions.

How to update Gaussian belief based on motions and observations?

Gaussians: Marginalization and conditioning

Linear model for motion and observation

Both models can be expressed through a linear function

A Gaussian which is transformed through a linear function stays Gaussian.

Linear models:

The KF assumes linear transition and observations

Zero means gaussian noise.

**xt = Atxt-1 + Bt ut + t  ....1(prediction step)**

**zt = Ct xt + t ....2(correction step)**

At – How a state is evolved from t-1 to t (without control or noise). It is an n\*n matrix

Bt – Describes how the control ut changes the state from t-1 to t. It is an n\*l matrix

t - Gaussian error

t – random variable

n – dimentionality of state space

l - dimensionality of control commands

Ct – Describes how to map a state xt to an observation zt. It is an k\*n matrix

k – state space

Rt – Noise for the motion

Qt – Noise for the control

Kalman filter assumptions:

* Gaussian distribution and noise
* Linear motion and observational model

What if this is not the case?

* Most realistic problems involve non- linear functions

Linear assumption revisited:

* For a linear function, a gaussian distribution squeezed through the linear function gives another gaussian distributions
* Whereas for a non-linear function, input gaussian is transformed into a non-gaussian function

Non-Gaussian Distributions

* The non-linear function leads to non-gaussian distribution
* Kalman filter is not applicable anymore

What can be done to resolve this?

**Local linearization**

**Extended Kalman Filter**

* Replacing linear model with non-linear model
* Extension of Kalman filtering
* One way to handle the non-linearities
* Performs local linearization
* Works well in practice for moderate. Non-linearities
* Large uncertainties leads to increased approximation errors