Kalman Filter

The Kalman filter model assumes the state of a system at time t evolved from the prior state at time t-1 per the equation

* , the state vector of the system (position, velocity, orientation)
* , the process noise, assumed to be zero mean multivariate Gaussian noisej

Kalman filter is a recursive estimator. Only the estimated state from the previous time step and measurement from the current time step are needed to compute the estimate for the current state. No history of observations nor estimates is needed.

The Kalman filter can be written as a single equation, however it is most often conceptualized as two distinct phases: “Predict” and “Update”.

1. **Predict**

* , the state vector
* , the state transition matrix
* , the control input matrix
* , the control input vector
* , the covariance matrix of the state vector terms
* , the process noise covariance matrix associated with noisy control inputs

1. **Measurement update**

* , the vector of measurements
* , the transformation matrix that maps the state vector parameters into the measurement domain
* , the covariance matrix of the measurement noise
* , the Kalman gain
* , the state vector following data fusion, i.e., the measurement update
* , the state vector before data fusion, i.e., the prediction
* , the covariance matrix (confidence) following data fusion
* , the covariance matrix (confidence) before data fusion

1-D Kalman filter without control

The math becomes much simpler for 1-D case without control inputs. Vectors and matrices become scalers.

1. Prediction

1. Measurement update

* *q*, the process noise
* *r*, the sensor noise

“The initial value for *p* is not very important since it is adjusted during the recursive calculation. It must be just high enough to narrow down. The initial value for the readout is also not very important for the same reason. But tweaking the values for the process noise and sensor noise is essential to get clear readouts.”