K

Kalman Filter

Gregory F. Welch
College of Nursing, Computer Science, Institute for Simulation & Training, The University of Central Florida, Orlando, FL, USA

Synonyms

Kalman-Bucy filter; KF

Related Concepts

▸ Sensor Fusion

Definition

The Kalman filter is a set of mathematical equations that provides an efficient computational (recursive) means to estimate the state of a process, in a way that minimizes the mean of the squared error. The filter is very powerful in several aspects: it supports estimations of past, present, and even future states, and it can do so even when the precise nature of the modeled system is unknown.

Background

In 1960, Rudolf E. Kalman published his famous paper describing a recursive solution to the discrete-data linear filtering problem [1]. Since that time, due in large part to advances in digital computing, the Kalman filter has been the subject of extensive research and application, particularly in the area of autonomous or assisted navigation. The goal of the filter is to produce evolving optimal estimates of a modeled process from noisy measurements of the process.

Theory

The Kalman filter addresses the general problem of trying to estimate the state $x \in \mathbb{R}^n$ of a discrete-time controlled process that is governed by the linear stochastic difference equation

$$x_k = Ax_{k-1} + Bu_{k-1} + w_{k-1} \qquad (1)$$

at time step k, with a measurement $z \in \mathbb{R}^m$ that is

$$z_k = Hx_k + v_k. \qquad (2)$$

The random variables w_k and v_k represent the *process noise* and *measurement noise*, respectively. They are assumed to be independent of each other, white, and with normal probability distributions

K. Ikeuchi (ed.), *Computer Vision*,
https://doi.org/10.1007/978-3-030-03243-2_716-1

$$p(w) \sim N(0, Q), \text{ and} \qquad (3)$$

$$p(v) \sim N(0, R). \qquad (4)$$

The $n \times n$ matrix A in the difference Eq. (1) relates the state x at the previous time step $k-1$ to the state x at the current step k, in the absence of either a driving function or process noise. The $n \times l$ matrix B relates an optional control input $u \in \mathbb{R}^l$ to the state x. The $m \times n$ matrix H in the measurement Eq. (2) relates the state to the measurement z_k.

One usually does not know the true form of the process (1) and associated noise parameter (3) nor the true measurement model (2) and associated noise parameter (4), but in practice one can often arrive at useful models via analytical formulations and laboratory-based measurements.

Using the process and measurement models (1)–(4), and real (noisy) measurements $\hat{z}_k$ at each time step k, the *Kalman filter* is used to recursively estimate the first two statistical moments of the process: the mean $\hat{x}_k$ and the error covariance P_k.

The filter is typically implemented in two steps, a *time update* step and a *measurement update* step, as follows:

Time update:

$$\hat{x}_k^- = A\hat{x}_{k-1} + Bu_{k-1}$$

$$P_k^- = AP_{k-1}A^\mathsf{T} + Q$$

Measurement update:

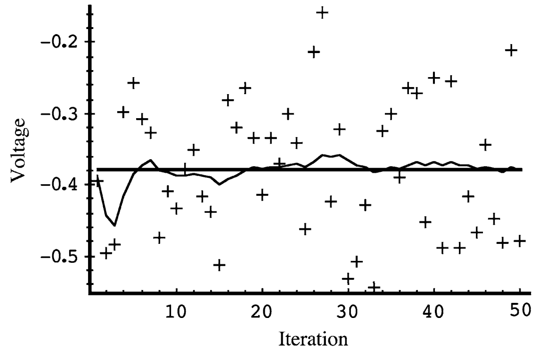
$$K = P_k^- H^\mathsf{T}(HP_k^- H^\mathsf{T} + R)^{-1}$$

$$\hat{x}_k = \hat{x}_k^- + K(\hat{z}_k - H\hat{x}_k^-)$$

$$P_k = (I - KH)P_k^-$$

Repeatedly applying these steps recursively estimates the process mean $\hat{x}_k$ and the error covariance P_k. Because the measurements can vary in form and timing, the filter is often characterized as a tool for *sensor fusion*.

The *Kalman filter* is optimal in that the $n \times m$ *Kalman gain* matrix K minimizes the trace of a posteriori error covariance P_k.

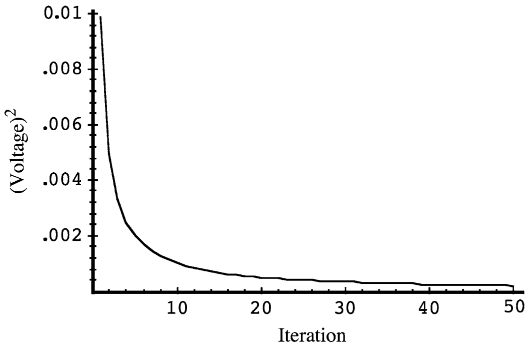


Kalman Filter, Fig. 1 An example: estimating a random constant from noisy measurements. The true random constant x_k (*solid line*), the noisy measurements $\hat{z}_k$ (*cross marks*), and the filter estimate $\hat{x}_k$

An accessible high-level introduction to the general idea of the Kalman filter can be found in Chap. 1 of [2]. A more complete introduction can be found in [3] and in [4] which also contains some interesting historical narrative. More extensive references include [2, 5–9].

Application

Despite the fact that employed process models rarely match the corresponding true systems, and the noise models rarely exhibit the characteristics required for optimality (zero mean, normally distributed, and independence over space and time), the Kalman filter remains popular – perhaps due to its relative simplicity and robustness. It continues to be used widely in diverse application areas such as electronics, robotics, localization, navigation, and even economics. In computer vision, variations of the Kalman filter are typically used to estimate structure, motion, and camera parameters. Early examples include [10–13]. Both the OpenCV software project [14, 15] and the Matlab numerical computing environment [16] include Kalman filter functions.



Kalman Filter, Fig. 2 The error covariance P_k associated with the estimates in Fig. 1. After 50 iterations, the covariance has settled to a relatively small 0.0002 volts^2

Experimental Results

A relatively simple example of using the Kalman filter to estimate a scalar random constant is given in [3], with complete details for the structure of the filter, the parameters, the initial conditions, and various results. The example presumes access to noisy measurements of a voltage that is corrupted by a 0.1 volt RMS white measurement noise. Referring back to Eqs. (1) and (2), the value to be estimated is presumed constant so $A = 1$, there is no control input so $u = 0$ (and B is irrelevant), and the noisy measurements are of the state (the voltage) directly so $H = 1$. For a true voltage of $x = -0.37727$, $Q = 1 \times 10^{-5}$, and $R = (0.1)^2 = 0.01$, plots for the true voltage x_k, noisy measurements, and estimated voltage $\hat{x}_k$ are shown in Fig. 1; and the error covariance P_k is shown in Fig. 2.

References

1. Kalman RE (1960) A new approach to linear filtering and prediction problems. Trans ASME–J Basic Eng 82(Series D):35–45
2. Maybeck PS (1979) Stochastic models, estimation and control, vol 1. Volume 141 of Mathematics in science and engineering. Academic Press, New York
3. Welch G, Bishop G (1995) An introduction to the Kalman filter. Technical Report TR95-041, University of North Carolina at Chapel Hill, Department of Computer Science
4. Sorenson HW (1970) Least-squares estimation: from gauss to kalman. IEEE Spectrum **7** 63–68
5. Gelb A (1974) Applied optimal estimation. MIT Press, Cambridge, MA
6. Jacobs O (1993) Introduction to control theory, 2nd edn. Oxford University Press, New York
7. Lewis FL (1986) Optimal estimation: with an introduction to stochastic control theory. John Wiley and Sons, Inc., New York
8. Brown RG, Hwang PYC (2012) Introduction to random signals and applied kalman filtering with matlab exercises, 4th edn. Wiley & Sons, Inc, New York
9. Grewal MS, Andrews AP (2008) Kalman filtering theory and practice using MATLAB, 3rd edn. Information and system sciences series. John Wiley & Sons, Inc., New York
10. Stuller J, Krishnamurthy G (1983) Kalman filter formulation of low-level television image motion estimation. Comput Vis Graph Image Process 21(2):169–204
11. Matthies L, Kanade T, Szeliski R (1989) Kalman filter-based algorithms for estimating depth from image sequences. Int J Comput Vis 3(3):209–238
12. van Pabst JV, Krekel PFC (1993) Multisensor data fusion of points, line segments, and surface segments in 3D space. In Schenker PS (ed) Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series. Volume 2059 of Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, pp 190–201
13. Azarbayejani A, Pentland A (1995) Recursive estimation of motion, structure, and focal length. IEEE Trans Pattern Anal Mach Intell 17(6):562–575
14. Bradski G (2000) The OpenCV Library. Dr. Dobb's Journal of Software Tools
15. OpenCV (2019) Open source computer vision library (opencv). http://opencv.org
16. Mathworks T (2019) Matlab. https://www.mathworks.com/products/matlab.html