

Target Tracking Using Kalman Filter*

Prasad Kalane
PREC Loni
Pune University

ABSTRACT

One of the fundamental problems in vision is that of tracking target through sequences of images. Within this paper I discuss the design of Kalman filter algorithm to track the target and shows the resulting improvement in tracking. This is of utmost importance for high-performance real-time applications. More specifically, I describe the technique of how to track Moving Pen. Experimental results demonstrate that, the proposed algorithm is more robust in tracking performance. The proposed method is suitable for indoors as well as outdoors scenes with static background and overcomes the problem of stationary targets fading into the background[#].

Keywords

*Rudolf E. Kálmán introduced this techniques in 1960 [Kalman, R.E.: "A New Approach to Linear Filtering and Prediction Problems", Transactions of the ASME - Journal of Basic Engineering Vol. 82: pp. 35-45] & in 1961 along with Richard S. Bucy [Kalman, R.E., Bucy R.S.: "New Results in Linear Filtering and Prediction Theory", Transactions of the ASME - Journal of Basic Engineering Vol. 83: pp. 95-107 (1961)], so also called as Kalman-Bucy filter.

[#] Commonly found in variations of Stauffer's adaptive background algorithm, [C. Stauffer and W. E. L. Grimson, "Learning patterns of activity using real-time tracking", IEEE Trans. on Pattern Anal. and Machine Intel., Vol. 22, No. 8, pp. 747-757, 2000.].

Kalman filter, median filter. Measurement update, prediction.

1. INTRODUCTION

Target tracking can be described as the process of determining the location of a target feature in an image sequence over time. It is one of the most important applications of sequential state estimation, which naturally admits Kalman filter and particle filter as the main candidate. It capture significant attention during the past several years due to its crucial value in visual applications including augmented reality, surveillance perceptual user interfaces, object-based video compression, driver assistance smart rooms & smart highways, etc. In recent years there has been much work on the tracking of moving objects within a scene. Systems developed for such tasks as people tracking[7][11] face tracking[1]and vehicle tracking[22][23] have come in many shapes or size[24].

In this paper, a moving pen tracking method is proposed.

2. Target tracking and associated problems

Nathan Funk discussed the following problem associated with tracking[25]. The target to be tracked might be a complete object (e.g. a person) or a small area on an object (e.g.hand of a person). In either case, the feature of interest is typically contained within a target region. Ideally, a Tracking Algorithm would be able to locate the object anywhere within the image at

any point in time. However typically only a limited region of the image is searched. Reasons for this are efficiency (especially necessary for real-time applications) and the fact that there might be many other similar-looking objects in the image.

The intuitive approach is to search within a region centered around the last position of the object. But as Figure 1 illustrates, this approach will fail if the object moves outside the target range.

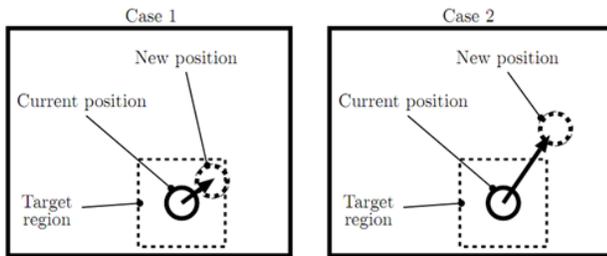


Figure 1: Case 1: Tracking without position prediction might be successful,
Case 2: Tracking without prediction fail.

There are many possible reasons why the object might not stay within this region:

- The object is moving too fast.
- The frame rate is too low.

In summary, two major problems have been identified:

1. The object can only be tracked if it does not move beyond the searched region.
2. Various factors such as lighting and occlusions can affect the appearance of the target, thus making accurate tracking difficult.

To solve this problem, we can attempt predicting the location of the target, and searching in a region centered around that location. The position prediction method needs to be robust enough to handle the source of error, as the previously obtained location

measurements are likely not accurate. This paper will discuss how these problems can be addressed with the Kalman filter, and how well they are solved.

There are two key steps in target tracking:

1. Detection of interesting moving objects (Background Modelling)
2. Tracking of such objects from frame to frame.

3. Background Modelling

Because target can be considered movable elements in the environment, it is very helpful to separate the points that belong to the environment (back-ground) from those that do not (foreground)[26].

The background information is important for two reasons.

1. If some of the target features are also present in the background;
2. In many applications it is difficult to exactly delineate the target, and its model might contain background features as well.

3.1 Related works:-

The targets may be either person, or vehicles. The common property of these targets is that sooner or later they must exhibit some movement which is evidence that distinguishes them from the background and identifies them as foreground target. The segmentation of foreground objects can be accomplished by processing the difference of the current frame from a background image. This background image can be static[1] or can be computed adaptively[2]. The drawback of the static background image is that background does change. *In outdoor scenes:* Natural light conditions change slowly as time goes by; the wind causes swaying movements of flexible

background object (e.g. foliage). *In indoor scenes*: fluorescent light flickers at the power supply frequency; pieces of furniture are moved around (e.g. objects on tabletops and small pieces of furniture are rearranged.). The above background generation is straight forward and can be made to run very fast. Nevertheless, it is very limited in its application, being vulnerable to scenes in which there are many moving objects, or objects move slowly.

An altogether more robust algorithm[3] that compensates for these drawbacks is popularly used when the simple one fails. Also, all such effects can be learned by Stauffer's adaptive background algorithm[4] and any of its modifications[5],[6]. The drawback of all the existing variations of Stauffer's algorithm is that[7] stationary foreground objects tend to fade in the background. Small rates fade foreground objects slowly, but are also slow in adapting to the background changes, (e.g. the motion of a chair). Large rates favor background adaptation but tend to fade a target into the background when it stops. This fading progressively destroys the region of the tracked object, deforms its perceived shape and finally leads to losing track of the object altogether. When the target resumes moving, foreground pixels will be marked only at the locations not previously occupied by the stationary target. When the target has fairly uniform coloration, this can lead to track loss even in the presence of movement. This method does not work in most practical situations due to noise in the intensity, caused by imaging and lighting effects, and spatial noise, caused by camera jitter.

Aristodemos Pnevmatikakis and Lazaros Polymenakos^[7] propose a novel tracking system that addresses many of the above mentioned limitations by utilizing a feedback mechanism from the tracking module to the adaptive background module which in turn provides the evidence for each target to the tracking module. There have been many methods proposed for the modelling of backgrounds, some of them

are[24]

Baumberg and Hogg[8] use a median filter at each pixel to construct a background model. A threshold absolute image difference is used to identify foreground pixels.

Haritaoglu et al.[9] Model background pixel intensities using minimum and maximum values in addition to a maximum difference between frames. If pixel intensity falls outside this model it is classified as foreground.

Ridder et al.[10] Use a Kalman filter at each pixel to model pixel intensities and predict a single value background model, however the foreground detection is performed using a threshold absolute image difference.

Darrell et al[13] use a height map of the environment (built using stereo information) for background model to estimate the trajectory of moving people in a room.

Wren et al.[11] Model pixel colour as a full covariance Gaussian distribution in YUV colour-space. Similarly, McKenna et al.[12] Use a diagonal covariance Gaussian in RGB colour-space.

3.2 Our approach:-

We chose to use a simple median filter to remove non hand components of the binary image since our background was not too complicated. As we are tracking Pen assuming a stationary camera position and constant illumination, we will use Baumberg and Hogg[15] approach to construct a background model..

The simplest approach to modeling background is to consider the first frame to be the background, and to subtract intensities pixel by pixel in all successors. Non-zero differences then represent movement;

$$d(i, j) = 0 \dots\dots\dots \text{no motion}$$

$$= 1 \dots\dots\dots \text{motion detect}$$

This approach detects every single

movement, e.g. wind shaking trees, and the tiniest camera movement. More subtly, small changes in lightening condition would have catastrophic effects. We took this approach under a very strict control of an absolutely rigid camera with no environmental variation in lightening or shadow, and minimal or absent moving 'clutter'.

4. Tracking the Target

It consist of,

- (i) predicting their future positions according to its past movement using the Kalman filter and,
- (ii) When a foreground object is identified as a Target here pen, the system starts to track it.

4.1 Kalman Filter

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[27].

The Kalman filter is the best filter among the subset of all linear filters and the best filter among the set of all filters when the noise processes are Gaussian type[21].

The Kalman filter is essentially a set of mathematical equations that implement a predictor-corrector type estimator that is optimal in the sense that it minimizes the estimated error covariance-when some presumed conditions are met [28].

Kalman filtering, in the spirit of Kalman filter[17] or Kalman-Bucy filter[19]

consists of an iterative prediction-correction process (see Figure 2)

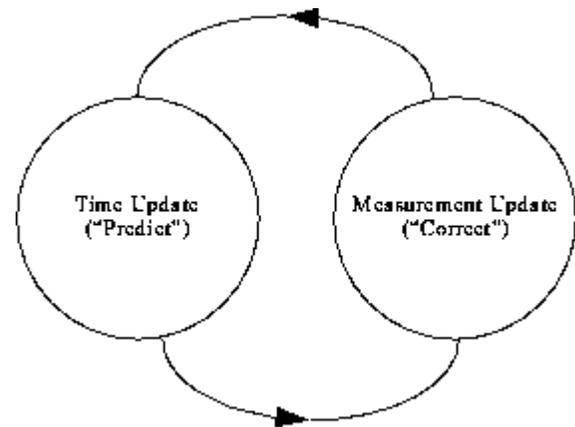


Figure 2. The ongoing discrete Kalman filter cycle. The time update projects the current state estimate ahead in time. The measurement update adjusts the projected estimate by an actual measurement at that time[18]

In the prediction step, the time update is taken where the one-step ahead prediction of observation is calculated. (i.e. The time update projects the current state estimate ahead in time)

In the correction step, the measurement update is taken where the correction to the estimate of current state is calculated (i.e. The measurement update adjusts the projected estimate by an actual measurement at that time).

4.2 Kalman Filter Design:

I used following description of Kalman filter by Welch & Bishop[27][28].

The equations for the Kalman filter fall into two groups: time update equations and measurement update equations.

The time update equations:

The time update equations are responsible for projecting forward (in time) the current state and error covariance estimates to obtain the a priori estimates for the next time step.

The measurement update equations:

The measurement update equations are responsible for the feedback—i.e. for incorporating a new measurement into the a priori estimate to obtain an improved a posteriori estimate.

The first task during the measurement update is to compute the Kalman gain. After each time and measurement update pair, the process is repeated with the previous a posteriori estimates used to project or predict the new a priori estimates. This recursive nature is one of the very appealing features of the Kalman filter—it makes practical implementations much more feasible than (for example) an implementation of a Wiener filter[20], which is designed to operate on all of the data directly for each estimate. The Kalman filter instead recursively conditions the current estimate on all of the past measurements.

Figure, below offers a complete picture of the operation of the filter,

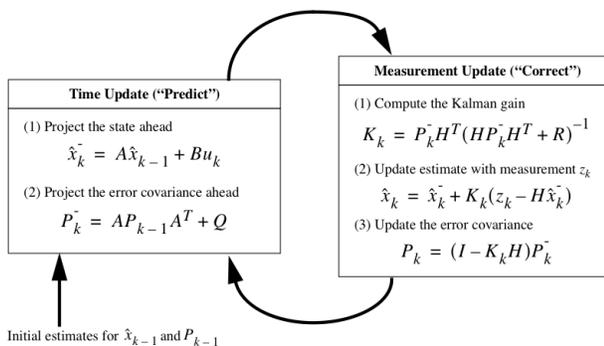


Figure 3: A complete picture of the operation of the Kalman filter[27][28]

Depending on the application one might want to obtain an estimate of the state at a certain time. If the state is estimated for some future time, the process is called prediction. If the estimate is made using all measurements up to and including the current moment, one speaks of filtering. If an estimate is made for some time in the past using measurements until the current moment, the process is called smoothing[29].

In practice, the use of Kalman filter is limited by the ubiquitous nonlinearity and non-Gaussianity of physical world.

Nathank Funk[25] discuss the robustness tracking using Kalman Filter as follows:

The main result of his experiment is the RMS errors of the individual image sequence/prediction method combinations. They are listed in the following table:

		Simple Prediction	Kalman Prediction
Constant Velocity	RMS Prediction Error	0.513	0.699
	RMS Measurement Error	0.113	0.128
Constant Acceleration	RMS Prediction Error	0.435	0.129
	RMS Measurement Error	0.032	0.026

An interesting observation that can be made from these numbers is that in the constant velocity case, the simple prediction method is more accurate than the Kalman filter. In the first images, both prediction methods show a considerable error. This is due to the fact that both prediction methods have no previous positions to derive a prediction from, so they predict that the target will stay at the original location. After the second image, they “become aware” of the motion, and the predictions improve. But the simple prediction method adapts quicker since it only bases its prediction on the two previous positions. The Kalman prediction method however takes longer to recover from these initial errors.

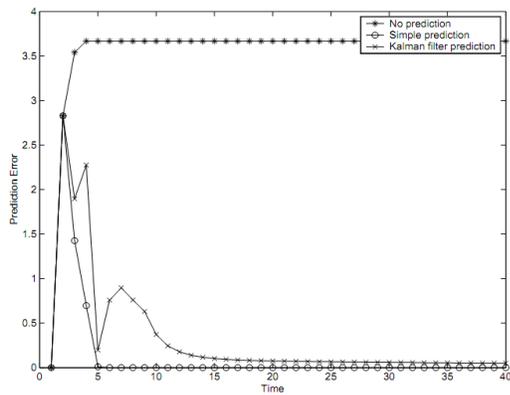


Figure 4: Comparison of the prediction errors for the constant velocity case. The simple prediction method predicts the position most accurately. The Kalman filter predictions are accurate only after the 10th image

For the constant acceleration case however, the Kalman filter clearly outperforms the simple prediction method after 3 images as is shown above. Since the simple prediction method does not expect the increasing velocity, it always lags behind the target with an approximately constant error. The Kalman filter reduces the prediction error quickly, and after 5 frames the prediction error stays below 0.1. This difference is also reflected in the RMS error values.

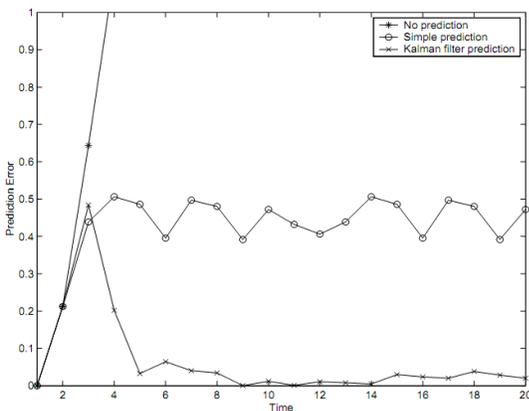


Figure 5: Comparison of the prediction errors for the constant acceleration case. This figure shows that the Kalman filter predicts the position of the target accurately starting with the fifth image.

The first hypothesis states that the

results of the Kalman filter are not better than the simple prediction method for cases without noise. The experiments however showed that this is not the case. Two reasons have been discovered: (1) The Kalman filter does not recover as quickly as the simple prediction process from the effects of the initial error. (2) The Kalman filter predicts more accurately for the constant acceleration case. Hence, the hypothesis is definitely not true since the simple prediction method does not handle cases with accelerated motion well.

5. EXPERIMENT & RESULT

During the explanation of the model we have shown examples of its performance. A broader experimentation has been done to test tracking of pen under different velocity and different distances from the vision system. Pen tracking result are shown below.



Figure 6: Background Image



Figure 7: Pen image 1

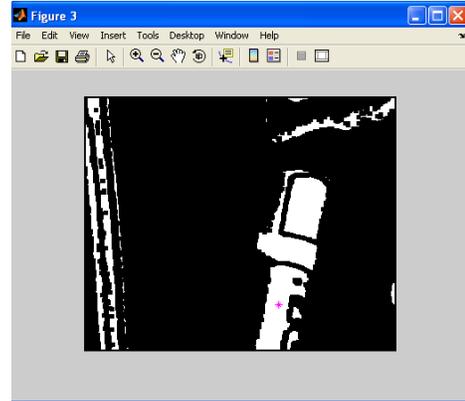


Figure 10: Tracked image of Pen 2

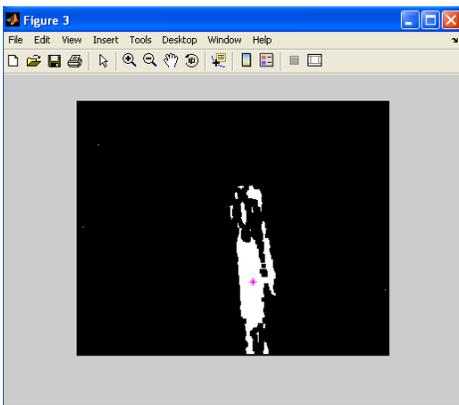


Figure 8: Tracked image of Pen 1



Figure 9: Pen image2

6. CONCLUSION & FUTURE WORK

We have presented a system able to detect and track pen. Therefore, as future work we also planning to track our own vehicle with respect to other vehicle to avoid major accidents (the "smart" highway systems that will eventually orchestrate the smooth movement of cars at high speeds in tightly bunched (read: tailgating) groups known as platoons). Car collision avoidance is very similar to the target tracking problem, here we are interested in predicting the own car's and other object's future position [Stephen Rohr, Richard Lind, Robert Myers, William Bauson, Walter Kosiak, and Huan Yen's, "An integrated approach to automotive safety systems". Automotive engineering international, September 2000]. Based on the prediction, collision avoidance actions such as

warning,braking and steering are undertaken when a collision is likely to happen. In order to have enough time to warn the driver the prediction horizon needs to be quite long. Therefore, utilizing knowledge about road geometry and infrastructure becomes important. One way to improve the prediction of possible manoeuvres is to use information in a digital map [Fredrik Gustafsson, Fredrik Gunnarsson et. al's "Particle Filter for Positioning, Navigation,

& Tracking” Dept. of Electrical Engineering, Sweden.(2001)].

The tracking solution presented in this paper has several desirable properties: *it is efficient, modular, has straightforward implementation, and provides superior performance on most image sequences.*

7. ACKNOWLEDGMENTS

This paper would not have been possible without the contributions of numerous researchers in this ever growing field. I would like to thank all of them – especially Prof. **Gregory F. Welch**, (Research Professor, Institute for Simulation & Training, Department of Electrical Engineering & Computer Science, The University of Central Florida) for his kind advice.

8. REFERENCES

- [1] H. Ekenel and A. Pnevmatikakis, “Video-Based Face Recognition Evaluation in the CHIL Project”, Int. Conf. Pattern Recognition, 2006.
- [2] A. McIvor’s, “Background Subtraction Techniques”.
- [3] C. Stauffer and W. E. L. Grimson “Adaptive background mixture models for real-time tracking”, In CVPR’99:Computer Society Conference on Computer Vision and Pattern Recognition, Ft. Collins, USA, volume 2, pages 246-252. 1999
- [4] C. Stauffer and W. E. L. Grimson, “Learning patterns of activity using real-time tracking”, IEEE Trans. on Pattern Anal. and Machine Intel., Vol. 22, No. 8, pp. 747–757, 2000.
- [5] P. KaewTraKulPong and R. Bowden, “An Improved Adaptive Background Mixture Model for Real-time Tracking with Shadow Detection”, in Proc. 2nd European Workshop on Advanced Video Based Surveillance Systems (AVBS01), Sept 2001.
- [6] J. L. Landabaso and M. Pardas, “Foreground regions extraction and characterization towards real-time object tracking”, in Proc. of Joint Workshop on Multimodal Interaction and Related Machine Learning Algorithms (MLMI ’05), 2005.
- [7] Aristodemos Pnevmatikakis and Lazaros Polymenakos “2D Person Tracking Using Kalman Filtering and Adaptive Background Learning in a Feedback Loop”.
- [8] A. Baumberg and D. Hogg. “Learning flexible models from image sequences”.In Eklundh J.O.,editor, 3rd European Conference on Computer Vision,Stockholm, Sweden, pages 299–308, Berlin, Springer Verlag, 1994
- [9] I. Haritaoglu, D. Harwood, and L. Davis. W4 “Who? When? Where? a real time system for detecting and tracking people”. In Proc. IEEE Conference on Automatic Face and Gesture Recognition, pages 222–227, 1998.
- [10] C. Ridder, O. Munkelt, and H. Kirchner . “Adaptive background estimation and foreground detection using kalman-filtering”. In Proc. International Conference on Recent Advances in Mechatronics, pages 193–199, 1995.
- [11] C. Wren, A. Azarbayejani, T. Darrell, and A. Pentland. Pfinder: “Real-time tracking of the human body”. IEEE Transactions on PAMI, 19(7):780–785, 1997]
- [12] S. McKenna, S. Jabri, Z. Duric, and H. Wechsler. “Tracking interacting people”. In Proc. International Conference on Automatic Face and Gesture Recognition, pages 348–353, 2000.
- [13] T. Darrell, D. Demirdjian, N. Checka, and P. Felzenszwalb. “Plan-view trajectory estimation with dense stereo background models”. In Eighth IEEE International Conference on Computer Vision (ICCV

- 2001), volume 2, pages 628 – 635, 2001.]
- [14] McFarlane N.J.B. and Schofield C.P. “Segmentation and tracking of piglets in images”. *Machine Vision and Applications*, 8:187-193, 1995
- [15] A.M. Baumberg and D.C. Hogg, “An efficient method of contour tracking using active shape models”. In *Proceedings of the IEEE Workshop on Motion on Non-rigid and Articulated Objects*, pages 194-199, Texas, 1994]
- [16] M.S. Grewal and A.P. Andrews “Kalman Filtering. Theory and Practice”. Prentice Hall, 1993
- [17] R. E. Kalman, “A new approach to linear filtering and prediction problem” *Trans. ASME, Ser. D, J. Basic Eng.*, vol. 82, pp.34–45, 1960.
- [18] <http://www.cs.brown.edu/stc/education/course95-96/Kalman-Filters/kalman.html>
- [19] R. E. Kalman and R. S. Bucy, “New results in linear filtering and prediction theory,” *Trans. ASME, Ser. D, J. Basic Eng.*, vol.83, pp. 95–107, 1961.
- [20] Brown, R. G., & Hwang, P. Y. C. (1996). “Introduction to Random Signals and Applied Kalman Filtering: with MATLAB Exercises and Solutions” (Third ed.): Wiley & Sons, Inc])
- [21] Anderson, B.D.O.and J.B. Moore “Optimal Filtering. Prentice-Hall, Englewood-Cliffs, NJ (1979)
- [22] D. Beymer, P. McLauchlan, B. Coifman, and J. Malik. “A real-time computer vision system for measuring traffic parameters”. In *Proc. CVPR*, pages 495–501, 1997
- [23] G.D. Sullivan. “Model-based vision for traffic scenes using the ground-plane constraint”. *Real-time Computer Vision*, 1994. D. Terzopoulos and C. Brown (Eds.). CUP
- [24] D R Magee, “Tracking Multiple Vehicles using Foreground, Background and Motion Models”, December 2001 Submitted to *European Conference on Computer Vision*, May 2002
- [25] Nathan Funk, “A Study of the Kalman Filter applied to Visual Tracking”, University of Alberta, Project for CMPUT 652, December 7, 2003
- [26] Rafael Muñoz-Salinas, Eugenio Aguirre, Miguel García-Silvente, “People Detection and Tracking using Stereo Vision and Color”, DECSAI-SI-2005-04
- [27] Greg Welch and Gary Bishop “An Introduction to the Kalman Filter” TR 95-041, Department of Computer Science University of North Carolina at Chapel Hill Chapel Hill, NC 27599-3175 Updated: Monday, July 24, 2006
- [28] Greg Welch and Gary Bishop “An Introduction to the Kalman Filter” University of North Carolina at Chapel Hill, Department of Computer Science Chapel Hill, NC 27599-3175, SIGGRAPH 2001 Course 8
- [29] M. A. SALZMANN “SOME ASPECTS OF KALMAN FILTERING’ Department of Geodesy and Geomatics Engineering University of New Brunswick August 1988, Latest Reprinting February 1996 August 1988