# Kalman-Filter - testing and evaluation

Marlon Lückert

June 5, 2020

# 1  Introduction

In order to evaluate if and how much better our filter implementation performs compared to the standardized Kalman-Filter we have to run tests and find a way to objectively compare the results. In the next sections we are going to discuss the test environment and the data we acquire from it.

# 2  Testing

Our Kalman-Filter is specifically designed to track and predict the position of billiard balls. The final goal is to implement this filter in a mobile app, which analyzes video footage preferably in realtime.

To test the filter we compare calculated position of the filter to the real position of the billiard pool.

But we cannot test our filter with video material because we do not know the ground truth of the balls' position in the video. So we are not able to make statements about the quality of the filter if we cannot compare the filter result to the optimal result.

Therefore we have to develop a virtual test environment where we can control every parameter of the billiard game and know the actual values of the balls' position.

## 2.1  Simulation

The simulation has to reflect all characteristics and physical conditions of a real billiard table. The virtual environment consists of a billiard ball and four cushions. We can control the start acceleration of the ball where we can reflect start shots with different intensity. When the ball starts moving a constant deceleration is applied to mimic the friction of the table's fabric. If the ball touches a cushion the direction of movement is mirrored to reflect a collision.

Our simulation does not include the rotation of the ball which would leads to a different behavior on cushion collisions. We have to assume that the ball is hit directly in the middle with no rotation.

With the simulation we can access all parameter of the ball like the position, speed and acceleration.

## 2.2 Modifying the simulation

The purpose of a Kalman-filter is to improve noised sensor data and filter bad measurements. When we analyze video footage the ball detection algorithm works as a sensor which provides the estimated position of the ball. The quality of this detection depends on different factors like the video resolution or the contrast of the images. But the sensor result will never be exactly the ground truth.

With the simulation we have perfectly correct values. So we have to modify the correct values by adding noise to reflect the real world circumstances. We have full control over the amount of noise, which makes it possible to simulate different qualities of sensors.

The simulation also provides data every time we want them. But in a real example you are limited to the framerate of the video and the performance of the ball detection algorithm. Therefore we have to define a sampling rate on which we process the data of the simulation.

## 2.3 Parameters for testing

After we defined the technical frame we can adjust different parameter to modify the simulation.

- start acceleration. How hard is the first hit of the ball?

- sensor noise. How close is the estimated position to the real position?

- sampling rate. How often is it possible to process a video frame?

We define three values for each parameter and compare them among each other which results in nine tests.

For the start acceleration we choose a slow hit with 80 cm/s, a medium hit with 1 m/s and a power shot with 4 m/s [1]

For the sensor noise we choose a deviation of 10 pixels, 50 pixels and 100 pixels.

For the sampling rate we choose 15 fps (very slow algorithm), 30 fps (realtime on typical mobile devices), 60 fps (realtime on modern devices).

Each of the nine tests are run with the standardized Kalman-filter and our filter implementation.

## 2.4 Data acquisation

With each test run we collect the estimated position with the normal Kalman-filter, our implementation and the ground truth. Additionally to the current position we collect the prediction of the position in 0.2 seconds, 1 second and 2 seconds.

# 3 Evaluation

To compare the results we calculate a single number which indicates the quality of each filter in each test. The indicator is the mean square error, which is calculated with the difference between the ground truth and the filter result in each frame of the testing simulation. After that we calculated the mean difference throughout the whole simulation. The smaller the mean square error the better performs the filter.

Since we are running multiple test with different parameters we can make statements about what parameter has the biggest influence of the quality of the filter.

# References

[1] "Typical pool ball speeds," Accessed: 05.06.2020.