Fusion of Heterogeneous Sensors for the Guidance of an Autonomous Vehicle

Jan C. Becker
Institute of Control Engineering
Technical University Braunschweig
Hans-Sommer-Str. 66
38106 Braunschweig, Germany
jan.becker@tu-bs.de

Abstract - This paper describes the sensor fusion system of an autonomous vehicle for automated vehicle testing. The vehicle sensor system for object-detection consists of a stereo vision sensor, four laserscanner (lidar) and a radar sensor. The sensor system is designed to totally cover the vehicle environment with a high redundancy in front of the vehicle. The sensor fusion system of the vehicle consists of a data alignment, a data association and a state estimation module. An adaptive information filter is used for the fusion of the associated targets from different sensors. The fused targets are input to the path planning and guidance system of the vehicle to generate a collision free motion of the vehicle.

Keywords: Sensor fusion, autonomous driving, data association, state estimation, information filter

1 Introduction

Autonomous driving has been subject to intense research in the past decades. Lately a variety of driver assistance systems have been introduced in commercial vehicles as a result of this research. Most of these systems such as anti-lock braking systems or vehicle dynamic control are based on inertial sensors, i.e. measuring the status of the vehicle itself. In recent years environmental sensors, e.g. video, radar, lidar and ultrasonic sensors, have been improved. Information about the status of the environment can now be incorporated into driver assistance functions such as adaptive cruise control or parking and navigation aids.

This project focuses on developing a completely autonomous driving vehicle which will be used for automated vehicle testing [10]. This application ranges at the very extreme of the spectrum of vehicle guidance functions.

The decentralized control system of this vehicle [3] uses information from a number of different sensors. These sensors can be classified into ego-position-detecting sensors and object-detecting sensors. A stereo vision sensor [6] detects the road lane markings and outputs the lane boundaries as a third-order polynomial. This information is combined with data from a DGPS sensor and a digital map to generate the accurate global position of the vehicle and the road geometry. Objects in the vehicle environment are detected by the vision sensor, four lasercanners and a radar sensor. Data from these sensors is fused in the sensor fusion system described in this paper.

The objective of the vehicle guidance system is to determine a collision free motion of the vehicle in consideration of the current vehicle state, information from a digital road map and the vehicle environment.

![Figure 1: Structure of the vehicle guidance system](image_url)
first step an ideal motion is planned without regarding detected obstacles. This motion consists of an ideal trajectory and an ideal velocity profile. The second step of the motion planning considers moving obstacles detected by the sensors and processed by the sensor fusion module. This provides the capability to react in real-time to a changing vehicle environment. The output of the vehicle guidance system is the desired motion of the vehicle. This motion is input to robust control algorithms that ensure that the vehicle follows the desired track even under the influence of disturbances. A robot driver [9] is used as a replacement of the human driver to operate the vehicle.

2 Multisensor Concept

In this paper, the sensor fusion system of an autonomous driving vehicle is presented. The objectives of this system are to create a unified description for the target data in this heterogeneous sensor environment and to accomplish an overall reduction of the large amount of sensor data for the subsequent systems of the autonomous vehicle as well as improving the accuracy of the estimated target states by combining information delivered by different sensors.

The autonomous vehicle is equipped with a stereo vision sensor, four laserscanner and a radar sensor for the detection and the tracking of obstacles that could potentially collide with the vehicle. The detection range of these sensors is overlapping in a large area (Fig. 2) in front of the vehicle, which is obviously the most important area for the vehicle path planning. For safety reasons this task is performed by a multitude of heterogeneous sensors.

This multisensor concept includes the following sensors:

- A stereo vision sensor is mounted behind the front screen. It is directed forwards with a viewing angle of 30° horizontally and 23° vertically and is aimed for mid and long range obstacles. This sensor simultaneously performs the tasks of object detection and lane recognition.

- Two laserscanner (Fig. 4) are mounted at the left and right end of the front bumper. Each sensor covers a range of 270°. The sensors overlap in the most important front viewing area. Further, a laserscanner with three laser beams (Fig. 5) is mounted in the center of the front and the rear bumper, respectively. These laserscanner are more robust to dynamic disturbances than single beam sensors. Thus the laserscanner system by itself covers a total look-around. The laserscanner are designed for short, mid and long range.
A long range radar sensor with a small viewing angle is mounted on the front bumper.

It can be seen that the degree of redundancy increases with the importance of the observed area. It is important that different wavelength and signal analysis principles are used by the sensors. The performance of each sensor system is dependent on environmental conditions such as weather or traffic situation. It must be accepted that the precision and reliability of each sensor is affected under unfavorable conditions. But it is crucial for automotive applications that each sensor notes its own limited performance and signals this state to the sensor fusion system. This sensor property is called self-assessment. In this sensor system self-assessment information such as confidence and reliability measures accompany each sensor measurement.

The next task in this multisensor environment is data alignment first in order to be ready for further processing. This received from the sensors has to be time synchronized unsynchronized from each other and target tracking of the detected object closed to the sensor as well as of characteristic points. These points consist of the point scanner used in this project describe an object by three parallel to the axes of the sensor. In contrast the laser-vision system describes an object by a cuboid that to-tion that fits best to its measurement principles. The kinematic variables. Each sensor uses a target descrip-

Figure 6: Structure of the sensor fusion module

Since the dissimilar sensors work independently and unsynchronized from each other and target tracking is performed by each sensor individually, the data received from the sensors has to be time synchronized first in order to be ready for further processing. This is called data alignment.

The data set of all sensor consists of position and kinematic variables. Each sensor uses a target description that fits best to its measurement principles. The vision system describes an object by a cuboid that totally surrounds the detected target with its sides parallel to the axes of the sensor. In contrast the laser-scanner used in this project describe an object by three characteristic points. These points consist of the point of the detected object closed to the sensor as well as of the outermost points to the left and right of the object. Such a compact object representation is necessary to guarantee real-time communication of the sensors with the fusion system.

The next task in this multisensor environment is to decide whether observations from different sensors actually represent the same target. This problem is called data association in this paper. The final task is the target state estimation or track data fusion. Here the corresponding targets are combined after it has been decided that two tracks actually represent the same target (Fig. 6).

Since the combined sensors may detect up to 90 objects during one sample period, a major constraint for the implementation of the fusion algorithms is that the system has to work online and in real time. Therefore good performance is just as important as a computationally efficient implementation of the algorithms.

3 Mathematical Description

3.1 Mathematical Model

For each (possibly dissimilar) sensor with the time be-tween sensor scans \(T\), the dynamics of targets are modeled as

\[
\mathbf{x}(k+1) = \mathbf{A}(k) \mathbf{x}(k) + \mathbf{g}(k) w(k),
\]

where \(\mathbf{x}(k)\) is the state vector at time \(k\) and \(\mathbf{A}(k)\) is the state transition matrix. The state vector consists of position and kinematic variables. In order to keep equations simple, the matrices shown in this paper include one dimension only. Thus the state transition matrix and the process noise transition vector are given by

\[
\mathbf{A}(k) = \begin{bmatrix}
1 & T(k) \\
0 & 1
\end{bmatrix} \quad \text{and} \quad \mathbf{g}(k) = \begin{bmatrix}
\frac{1}{2} T(k)^2 \\
T(k)
\end{bmatrix},
\]

respectively. The process noise \(w(k)\) accounts for the uncertainties in the model due to the acceleration of the target and is described by a white Gaussian random process with \(E\{w(k)\} = 0\) and \(E\{w^2(k)\} = \sigma_w^2(k)\). The measurement equation is given by

\[
\mathbf{y}(k) = \mathbf{C}(k) \mathbf{x}(k) + \mathbf{v}(k).
\]

The measurement noise \(\mathbf{v}(k)\) is described by \(E\{\mathbf{v}(k)\} = 0\) and \(E\{\mathbf{v}(k) \mathbf{v}^T(k)\} = \mathbf{R}(k)\).

3.2 Kalman Filter Algorithm

The data fusion strategy discussed in this paper is based on the discrete recursive Kalman filter, which is therefore briefly reviewed here. The prediction of the state vector \(\hat{\mathbf{x}}(k+1)\) calculated at time \(k\) for time \(k+1\) is based on the last estimate \(\hat{\mathbf{x}}(k)\):

\[
\hat{\mathbf{x}}(k+1) = \mathbf{A}(k) \hat{\mathbf{x}}(k) \tag{4}
\]

The prediction of the corresponding covariance matrix \(\hat{\mathbf{P}}(k+1)\) based on the previous covariance of the estimated state vector \(\hat{\mathbf{P}}(k)\) is determined by

\[
\hat{\mathbf{P}}(k+1) = \mathbf{A}(k) \hat{\mathbf{P}}(k) \mathbf{A}(k)^T + \mathbf{Q}(k) \tag{5}
\]

with

\[
\mathbf{Q}(k) = \mathbf{g}(k) \sigma(k)\mathbf{g}(k)^T. \tag{6}
\]
Then the innovation \( \nu(k+1) \), which is the difference between the actual measurement and the expected measurement, is determined.

\[
\nu(k+1) = y(k+1) - C(k+1)x'(k+1) \tag{7}
\]

The Kalman filter gain matrix is

\[
K(k+1) = P'(k+1)C^T(k+1)S^{-1}(k+1) \tag{8}
\]

with the innovation covariance matrix

\[
S(k+1) = C(k+1)P'(k+1)C^T(k+1) + R(k+1), \tag{9}
\]

where \( R(k+1) \) is the measurement noise matrix corresponding to the new sensor measurement \( y(k+1) \).

Then the state vector estimate \( \hat{x}(k+1) \) can be calculated using the new sensor observation \( y(k+1) \):

\[
\hat{x}(k+1) = x'(k+1) + K(k+1)\nu(k+1) \tag{10}
\]

In the last step, the covariance of the estimated state vector is updated from its predicted value:

\[
P(k+1) = [I - K(k+1)C(k+1)]P'(k+1) \tag{11}
\]

4 Data Alignment

The data from the dissimilar and unsynchronized sensors has to be time synchronized and transferred to a uniform coordinate system in order to be processed by the association and fusion algorithms. The prediction equations of the Kalman filter are implemented to predict sensor and covariance data from sensor measurement time \( l \) to the computing time \( k \) of the fusion module

\[
x'_m(k|l) = A_m(k-l)\hat{x}_m(l) \tag{12}
\]

\[
P'_m(k|l) = A_m(k-l)P_m(l)A_m^T(k-l) + g_m(k-l)\sigma^2 \tag{13}
\]

with the state transition matrix

\[
A_m(k-l) = \begin{bmatrix} 1 & (k-l)T_m \\ 0 & 1 \end{bmatrix} \tag{14}
\]

and the process noise transition vector

\[
g_m(k-l) = \begin{bmatrix} \frac{1}{2}(k-l)^2T_m^2 \\ (k-l)T_m \end{bmatrix} \tag{15}
\]

5 Data Association

This section deals with the process of associating observations from the sensors with each other. Since the target tracking is done by each sensor individually, the task of the data association unit in this sensor fusion module is to decide whether tracks from the different sensors actually represent the same target. The sensors are not only heterogeneous and dissimilar, but they are also mounted at different locations on the vehicle and thus have different viewing angles and perspectives onto the road and the observed obstacles. Therefore, further calculations in addition to the usual gating and target association algorithms are necessary.

The general procedure is as follows. Each sensor observation is classified by the sensor fusion module into one of the following three groups:

1. A target which has already been observed and tracked by the sensor at previous scans and is already associated with a track from another sensor.
2. A target which has already been observed and tracked by the sensor at previous scans but is not yet associated with a target detected by another sensor.
3. An object which is detected by this sensor for the first time.

A target which has already been associated with another track (case 1) is used to update the existing track. A reasonability check is performed to ensure that the association is still valid (track confirmation). This check includes the gating of the current position as well as velocity of both tracks. Only if all observations are within the gating regions, the association is kept. Otherwise, both tracks are released and new associations with other targets are possible.

Track data which is not yet associated with a track from another sensor (cases 2 and 3) is first passed through a gate in order to reduce the amount of possible track associations [4]. If an observation from a different sensor is within this (and only this) gate the observation is associated with this track. If an observation is not within any gate, the track is kept as a single track and a new association attempt will be performed during the next cycle. Different gating functions such as rectangular and ellipsoidal gates have been implemented and can be used depending on the covariance values of the track as well as on the density of existing tracks in the region of the gate.

If a target is within more than one gate or several targets from a single sensor are within the gate for one target from another sensor, the normalized distances between all targets are calculated and an assignment matrix is formed. If a possible target association is not within the formed gate, the corresponding entry in the assignment matrix is marked.

This matrix could be solved by enumeration or by more sophisticated algorithms which are in general more time-efficient. In this project an extended Munkres algorithm and a modified Nearest-Neighbor algorithm have been implemented.

The Munkres algorithm minimizes the sum of the distances of the associated targets and therefore obtains an optimal solution of the target-to-target assignment problem. The Munkres algorithm described in [5] has been modified to cope rectangular matrices, which are produced by sensors with an unequal number of observations. In addition, due to prior gating the assignment matrix generated by this system contains empty entries, which means that this association is excluded because one of the gating requirements has not been fulfilled. Thus the Munkres algorithm has also been extended in order to deal with matrices with empty entries in optimal time. This has led to a recursive implementation of the Munkres algorithm.

The modified Nearest-Neighbor algorithm obtains a sub-optimal solution. Associations with a single sensor observation are considered first in order to maximize
the total number of associations. Then the remaining matrix is solved by a Nearest-Neighbor criteria.

Even though the Nearest-Neighbor solution is in general sub-optimal, Monte Carlo simulations have shown that it is more than 10 times faster compared to the Munkres algorithm when solving a 10 by 10 assignment matrix. Using well-chosen gates, the solution of both algorithms barely differs, therefore the Nearest-Neighbor algorithm is used to satisfy the strong real-time requirements of the vehicle guidance system.

6 Data Fusion

6.1 Introduction

An information filter for state estimation [2, 7] can also be used for the fusion of data from more than one sensor. In this section, an approach for the application of the information filter to the problem of the fusion of multiple sensor data is described. A brief derivation of the information space and the information filter is given in the appendix of this paper. Finally, an approach for the adaptation of the process noise variance of the information filter by detecting target maneuvers is introduced.

6.2 Data Fusion with the Information Filter

In contrast to the Kalman filter, the information filter for multiple sensors shown in fig. 7 does not require the calculation of gain matrices or innovation covariance matrices. Instead it simply uses the amount of information contained in each sensor measurement. There is no direct state estimation in state space, instead the information of the distributed sensors on the desired state is fused. All information contributions are considered according to their quality.

The prediction and update equations of the information filter (eq. 38-41) have been derived in the appendix of this paper for a single sensor system. In this section the corresponding equations for a multiple sensor system will be stated.

The information contribution $i_i(k+1)$ and $I_i(k+1)$ of each observation $y_i(k+1)$ in a multi sensor system for the determination of a fused state vector can be defined by

$$i_i(k+1) \overset{\text{def}}{=} C_i^{-1}(k+1) \text{R}_i^{-1}(k+1) y_i(k+1)$$
$$I_i(k+1) \overset{\text{def}}{=} C_i^{-1}(k+1) \text{R}_i^{-1}(k+1) C_i(k+1).$$

The determination of the predicted estimates for the state vector and for the information matrix does not differ from the single sensor case and is therefore performed according to eq. 38 and 39 in the appendix:

$$\hat{z}_f(k+1) = \mathbf{Z}_f^*(k+1)A(k)\mathbf{Z}_f^{-1}(k) \hat{\mathbf{x}}_f(k)$$
$$\mathbf{Z}_f^*(k+1) = [A(k)\mathbf{Z}_f^{-1}(k)A^T(k) + \mathbf{Q}(k)]^{-1}$$

By using eq. 16 and 17 as well as by extending eq. 40 and 41 to the multi-sensor case one obtains the updated estimates for the multisensor information filter:

$$\hat{\mathbf{x}}_f(k+1) = \mathbf{Z}_f^*(k+1) + \sum_{i=1}^{n} i_i(k+1)$$
$$\mathbf{Z}_f(k+1) = \mathbf{Z}_f^*(k+1) + \sum_{i=1}^{n} I_i(k+1)$$

It can be seen that the fused information vector is a simple linear combination of the respective sensor information. The fused state vector is finally obtained by performing the transformation of $\mathbf{Z}_f(k+1)$ from information space back to state space:

$$\hat{\mathbf{x}}_f(k+1) = \mathbf{Z}_f^{-1}(k+1)\hat{\mathbf{x}}_f(k+1)$$

6.3 Adaptation of the Process Noise Variance

Usually target maneuvers, which are not covered by the kinematic model used, are modeled by the process noise, which is in general unknown. The unknown input can either be modeled as a random process or it can be estimated in real-time.

If maneuvers are modeled as a random process, a maximum acceleration in case of a constant velocity model has to be assumed. The drawback of this approach is, that a relatively large noise variance reduces the smoothing effect of the filter and thus results in an increase of the squared estimation error, i.e. a decrease
of the filter quality, while a relatively small noise variance may result in a slow reaction of the filter when a maneuver occurs.

In this approach the process noise variance $\sigma_w$ is adapted to detected target maneuvers in real-time. A simple procedure to detect target maneuvers is to calculate the normalized squared innovation of the filter:

$$
\epsilon_w(k) = \nu^T(k)S^{-1}(k)\nu(k)
$$

$\epsilon_w(k)$ is monitored and if it exceeds an upper limit $\epsilon_{max}$, the filter switches to a higher process noise value. Under a linear-Gaussian assumption, the probability density function of the normalized squared innovation is $\chi^2$-distributed with the dimension of the measurement. The limit is chosen according to the definition for a matched filter in [2] such that a maximum of 5% of all results of eq. 23 may be over the limit.

The variance of the filter can be adapted continuously or it can be switched between a number of discrete values. A continuous adaptation of the variance or an adaptation in a large number of discrete steps may lead to slower filter behavior, therefore only three variance levels are used for switching in this approach.

### 6.4 Simulations and Results

#### 6.4.1 Overview

In this section the multiple sensor information filter is compared to the Kalman filter based measurement fusion [8, 11]. To make the simulation results for both filter as comparable as possible, all filter equations have been implemented as written in this paper using Matlab. Triangularization and decomposition methods, which also exist for the information filter, are out of the scope for this paper.

The comparison of the two fusion concepts is performed by considering different sets of input signals. Three measurement vectors $v_1$, $v_2$ and $v_3$ of one target with possibly different measurement noise variances of 1, 80 and 240 for position and 1, 85, 256 for velocity variables, respectively, are fused. The filters are initialized with the first measurement of an arbitrary sensor.

#### 6.4.2 Comparison of Estimation Results

In the case where at least one measurement is very exact, i.e. the measurement variances are $\text{Var}(v_1) = 1$, $\text{Var}(v_2) = 80$ and $\text{Var}(v_3) = 240$, there is practically no difference in the results of two methods as expected. Assuming equal process models are used, the results are practically equivalent, independent of the disturbance. The low quality of two of the three measurement inputs has practically no influence on the result. Both filters strongly tend towards the high quality measurement. The advantages of the fusion of information will become obvious when all measurement inputs are of low quality, i.e. there is no dominant measurement of high quality. The course of the estimated variables is plotted over time in fig. 8 for a measurement set with one high quality measurement.

Fig. 9 shows, that the error peaks of the information filter are lower compared to the Kalman filter when the measurement set consists of only lower quality measurements. Each sensor information $y_i$ and $R_i$ enters the filter separately instead of being fused before the actual filtering process in case of the Kalman filter measurement fusion, where the averaged covariance matrix is in general too small when large amplitude changes in the measurement noise occur.

#### 6.4.3 Filter Consistency

No major differences in filter consistency between the two methods could be observed. The plot of the normalized squared estimation error in fig. 11a shows that the filter consistency for both methods is almost equal for the measurement set $\text{Var}(v_1) = 1$, $\text{Var}(v_2) = 80$, $\text{Var}(v_3) = 240$. For the measurement vector combina-
Position error variance

Velocity error variance

Figure 10: Resulting error variance after fusion

\[ \text{Var}\{v_1\} = 80, \text{Var}\{v_2\} = 80, \text{Var}\{v_3\} = 240 \]

Figure 11: Consistency test

\[ \epsilon_v \text{ for } \text{Var}\{v_1\}=1, \text{Var}\{v_2\}=80, \text{Var}\{v_3\}=240 \]

\[ \epsilon_v \text{ for } \text{Var}\{v_1\}=80, \text{Var}\{v_2\}=80, \text{Var}\{v_3\}=240 \]

7 Conclusions

In this paper, the system for the fusion of data from the object-detecting sensors of an autonomous vehicle has been presented. The focus of this article has been on the description of the implementation and the application of a sensor fusion system for a set of heterogeneous sensors. A modular concept for the fusion of heterogeneous sensors has been developed. After sensor data has been synchronized and aligned, the association of the targets of the different sensors is performed. The number of possible associations is reduced by using gates, then an assignment matrix is formed, which is solved by a modified Munkres or Nearest-Neighbor-algorithm. Finally, the data fusion and state estimation is performed by an information filter, which reacts to detected target maneuvers by an online adaptation of the process noise variance. First tests indicate a good performance of the sensors and the sensor fusion system. Further simulations and road tests will be carried out in order to validate these results.

Acknowledgment

The author would like to thank the Ministry of Economics, Technology and Transport of the State of Lower Saxony, Germany, the Volkswagen AG and the Robert Bosch GmbH for supporting this work.

References

Appendix: Derivation of the Information Filter Algorithm

The information space is completely defined by the information matrix, which is the inverse of the error covariance matrix for Gaussian distribution

\[ Z(k) \overset{\text{def}}{=} P^{-1}(k) \]  \hspace{1cm} (24)

and by the estimated information vector (state vector in the information space) as the product of \( Z(k) \) and the estimated state vector

\[ \hat{x}(k) \overset{\text{def}}{=} Z(k)x(k). \]  \hspace{1cm} (25)

The information filter [1, 7] can be derived by a transformation of the Kalman filter equations to the information space. Rearranging eq. 10 results in

\[ \dot{x}(k+1) = [I - K(k+1)C(k+1)]x'(k+1) + K(k+1)y(k+1). \]  \hspace{1cm} (26)

Then \( [I - K(k+1)C(k+1)] \) is expanded with the unity matrix expressions \( P'*(k+1) \) \( P'^{-1}(k+1) \) and \( S(k+1)S'^{-1}(k+1) \):

\[ I - K(k+1)C(k+1) = [P'^*(k+1) - K(k+1)C(k+1)P'^*(k+1)]P'^{-1}(k+1) \]
\[ = [P'^*(k+1) - K(k+1)S(k+1)S'^{-1}(k+1)]C(k+1)P'^*(k+1)]P'^{-1}(k+1). \]  \hspace{1cm} (27)

By using the transpose of eq. 8 in eq. 27

\[ I - K(k+1)C(k+1) = [P'^*(k+1) - K(k+1)S(k+1)K'^{T}(k+1)]P'^{-1}(k+1) \]  \hspace{1cm} (28)

is obtained. Using eq. 11, this results in

\[ I - K(k+1)C(k+1) = P(k+1)P'^{-1}(k+1). \]  \hspace{1cm} (29)

Combining eq. 8 and eq. 9 yields

\[ K(k+1) = P'(k+1)C'^{T}(k+1) \]
\[ (C(k+1)P'^*(k+1)C'^{T}(k+1) + R(k+1))^{-1} \]
\[ K(k+1) = P'^*(k+1)C'^{T}(k+1) \]
\[ P'^*(k+1)C'^{T}(k+1) + R(k+1))^{-1} \]
\[ K(k+1)R(k+1) = [I - K(k+1)C(k+1)] \]
\[ P'^*(k+1)C'^{T}(k+1) \]
\[ K(k+1) = [I - K(k+1)C(k+1)] \]
\[ P'^*(k+1)C'^{T}(k+1)R^{-1}(k+1) \]  \hspace{1cm} (30)

and by inserting eq. 29 into eq. 33 a different form of the filter gain matrix is obtained:

\[ K(k+1) = P(k+1)C'^{T}(k+1)R^{-1}(k+1) \]  \hspace{1cm} (34)

Substituting eq. 28 and eq. 34 into eq. 26 and multiplying the expression with \( P'^{-1}(k+1) \) results in the update equation for \( \hat{x}(k) \)

\[ P'^{-1}(k+1)\dot{x}(k+1) = \left[P'(k+1)P'^{-1}(k+1)\right]x'(k+1) \]
\[ + C'^{T}(k+1)R^{-1}(k+1)y(k+1). \]  \hspace{1cm} (35)

To obtain a solution for the estimation of the information matrix, the equations of the Kalman filter are used. Inserting eq. 28 and eq. 34 into eq. 11 yields

\[ P(k+1) = \left[P'(k+1)P'^{-1}(k+1)\right]x'(k+1) \]
\[ + C'^{T}(k+1)R^{-1}(k+1)y(k+1). \]  \hspace{1cm} (36)

By multiplying this equation twice with \( P'^{-1}(k+1) \), a simplified update equation for the information matrix is obtained

\[ P'^{-1}(k+1) = \left[P'^{-1}(k+1) + C'^{T}(k+1)R^{-1}(k+1)C(k+1) \right] \]
\[ \left[P'^{-1}(k+1) + C'^{T}(k+1)R^{-1}(k+1)C(k+1) \right]^{-1} \]  \hspace{1cm} (37)

Finally by using the definitions of the information space, the prediction equations

\[ z'(k+1) = Z'(k+1)A(k)Z'^{-1}(k+1)x(k) \]  \hspace{1cm} (38)
\[ Z'(k+1) = [A(k)Z'^{-1}(k+1)A'^{T}(k+1) + Q(k)]^{-1} \]  \hspace{1cm} (39)

and the update equations

\[ \hat{z}(k+1) = z'(k+1) + C'^{T}(k+1)R^{-1}(k+1)y(k+1) \]  \hspace{1cm} (40)
\[ z(k+1) = Z'(k+1) + C'^{T}(k+1)R^{-1}(k+1)C(k+1) \]  \hspace{1cm} (41)

of the information filter can be expressed.