Kalman Filter for Improved Radar Autocalibration Performances

Grégoire SANSOT[1,2] Romain NÉGRIER[1] Cyrille MENUDIER[1] Christophe LABARTHE[2]

[1]XLIM UMR CNRS 7252, University of Limoges, 87000 Limoges, France

[2]Thales LAS France, Rue Pierre Gilles de Gennes, 91470 Limours, France

Résumé – Les performances d'un système radar dépendent directement de la qualité de calibration du réseau d'antennes, puisque celle-ci influence directement la détection des cibles d'intérêt et donc leur pistage. Cependant, cette calibration peut se dégrader avec le temps à cause du vieillissement du matériel ou autres effets environnementaux, ce qui justifie l'intérêt porté aux méthodes d'autocalibration qui permettent de recalibrer le radar en cours de fonctionnement. Cet article présente une nouvelle approche utilisant un filtre de Kalman pour fusionner des estimations successives de gain, améliorant ainsi l'autocalibration. Cette méthode permet donc de s'affranchir d'un matériel de calibration coûteux et d'assurer la pérennité des performances du radar. La méthode présentée est validée par simulation.

Abstract – The performance of a radar system is dependent on the antenna array calibration, since this calibration has a direct influence on the detection of targets of interest, and hence on their tracking. However, the calibration can degrade over time due to aging components or other environmental effects, which justifies the interest in autocalibration methods that allow a radar to be recalibrated during operation. This article presents a new approach to autocalibration using a Kalman filter to merge successive gain estimates, thus improving autocalibration. This method eliminates the need for costly calibration equipment, and ensures that the radar's performance is maintained over the long term. The method presented is validated by simulation.

1 Introduction

Electronically steerable antenna arrays in radar systems are used to generate directional beams for target detection and tracking; their performance depends on precise calibration since even minor discrepancies can degrade the beamforming significantly, causing side-lobe level increases, distorted beam patterns, or even misaligned beams in extreme cases. Therefore, calibration is vital for optimizing performance and ensuring reliable operation.

Traditional methods like factory calibration methods [1, 2] provide accurate initial settings, but cannot account for component degradation over time. Online alternatives offer real-time adjustments, but are hardware-intensive and thus expensive, and similarly affected by component aging [3]. This highlights the need for more efficient solutions to reduce hardware reliance and cost while maintaining accuracy throughout the lifetime of the array.

Autocalibration techniques address these challenges by dynamically estimating and correcting errors during normal operation, without the need for designated hardware (other than good computation units) or extensive downtime [4, 5, 6, 7, 8]. However, these methods tend to lack accuracy. Additionally, most methods are designed to estimate the array's gain and phase error from a completely uncalibrated state. However, since the these errors can vary slowly over time, previous estimates must occasionally be discarded in favor of new ones to ensure sustained performances over time. This process is inefficient, as these estimates are typically highly correlated.

This article introduces a data fusion approach to autocalibration that leverages a Kalman filter to exploit correlations in successive complex gain estimates, enhancing precision without significantly increasing computational complexity as the Kalman filter is numerically efficient (see Fig. 1). This technique also allows the concurrent use of different calibration methods, and it also allows to calibrate only a subset of the array at a time which is particularly useful for large arrays or computationally intensive methods in real-time applications. Any autocalibration method that yields an estimate of the calibration vector without a steering direction bias can be used, which usually eliminates any method that does not use one or more calibration targets.

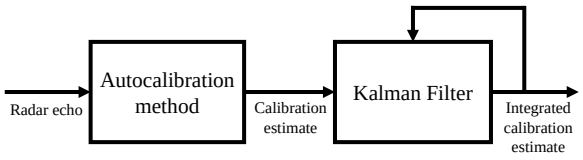


Figure 1: Block diagram of the Kalman fusion approach

2 Signal Model

In this paper, scalars are written in lowercase italic, vectors in lowercase bold and matrices are in capital bold.

Without loss of generality, let us consider an active 1-D antenna array of M elements with P impinging narrowband signals. The $M \times 1$ signal vector $\mathbf{x}(n)$ at time n is:

$$\mathbf{x}(n) = \text{diag}\left(\mathbf{g}\right)\mathbf{C}\mathbf{A}\left(\Theta\right)\mathbf{s}(n) + \mathbf{w}(n) \tag{1}$$

Here, $\mathbf{s}(n)$ is the vector of shape $P \times 1$ of the complex envelope of the P impinging signals at time n. $\mathbf{A}\left(\Theta\right)$ is the steering matrix of shape $M \times P$ for directions $\Theta = \begin{pmatrix}\theta_0 & \dots \theta_{P-1}\end{pmatrix}$ of the impinging signals. The vector $\mathbf{g}$ is the complex gain error of the array, and diag $(.)$ is the usual diagonal operator used in linear algebra. $\mathbf{C}$ is the $M \times M$ coupling matrix between elements of the array and $\mathbf{w}(n)$ is an unknown white gaussian noise vector.

The goal of autocalibration is to estimate $\mathbf{g}$. To remove ambiguity, a reference element r is chosen and its gain is arbitrarily chosen to be one: $g[r] = 1$.

The N time samples of a given scan are concatenated so that:

$$\mathbf{X} = \text{diag}\left(\mathbf{g}\right)\mathbf{C}\mathbf{A}\left(\Theta\right)\mathbf{S} + \mathbf{W} \tag{2}$$

with,

$$\mathbf{X} = \begin{pmatrix}\mathbf{x}(0) & \mathbf{x}(1) & \dots & \mathbf{x}(N-1)\end{pmatrix} \tag{3}$$

$$\mathbf{S} = \begin{pmatrix}\mathbf{s}(0) & \mathbf{s}(1) & \dots & \mathbf{s}(N-1)\end{pmatrix} \tag{4}$$

$$\mathbf{W} = \begin{pmatrix}\mathbf{w}(0) & \mathbf{w}(1) & \dots & \mathbf{w}(N-1)\end{pmatrix} \tag{5}$$

3 Kalman Filter for Calibration

In this section, the method to refine the calibration vector estimate over time using a Kalman filter is described [9]. The idea is that once the array is calibrated using a vector $\mathbf{h}$, the calibrated signal $\mathbf{X}_c$ is as follows:

$$\mathbf{X}_c = \text{diag}\left(\mathbf{h}\right)^{-1}\text{diag}\left(\mathbf{g}\right)\mathbf{C}\mathbf{A}\left(\Theta\right)\mathbf{S} + \mathbf{W} \tag{6}$$

with $(.)^{-1}$ the matrix inverse operator.

In an ideal world, $\mathbf{h}$ would be equal to $\mathbf{g}$, thus eliminating the complex gain error completely. However, that is almost never the case. Therefore, the remaining complex gain error could be estimated and corrected to improve performances. The calibration error vector to estimate at calibration step i is:

$$\mathbf{e}_i = \frac{\mathbf{g}}{\mathbf{h}_i} \tag{7}$$

Here, the division is an element-wise division. This is also the vector that is estimated using an autocalibration method. Let us define $\mathbf{y}_i$ the estimations of the complex calibration error vector at calibration step i using the chosen autocalibration method. Then, the following state-model is defined:

$$\begin{cases}\hat{\mathbf{e}}_i &= \mathbf{F}_i\hat{\mathbf{e}}_{i-1} + \mathbf{u}_i \\ \mathbf{y}_i &= \mathbf{D}_i\hat{\mathbf{e}}_i + \mathbf{v}_i\end{cases} \tag{8}$$

Here, $\hat{\mathbf{e}}_i$ is the Kalman estimate of the calibration error vector $\mathbf{e}_i$ at calibration step i. $\mathbf{F}_i$ is the evolution matrix, and $\mathbf{D}_i$ is the measurement matrix at calibration step i.

The vector $\mathbf{u}_i$ is the process noise, which is due to the deviation of the true calibration vector $\mathbf{g}$ over time. This noise is assumed to be white and gaussian with correlation matrix $\mathbf{Q}_i$. Finally, $\mathbf{v}_i$ is the measurement noise vector, which is dependent on the characteristics of the chosen autocalibration algorithm. This noise is assumed to be white and gaussian with correlation matrix $\mathbf{R}_i$.

The matrices $\mathbf{F}_i$, $\mathbf{D}_i$, $\mathbf{Q}_i$ and $\mathbf{R}_i$ are discussed later in this paper.

From this state model, the usual Kalman filter update and prediction steps can be taken to estimate the state vector $\hat{\mathbf{e}}_i$, as shown in Alg. 1. It should be noted that the meaning of the infinite loop in this algorithm is that these Kalman steps should be done regularly throughout the lifetime of the radar, not that all computing resources should be dedicated to this algorithm.

Here $(.)^H$ is the hermitian transpose operator, and $\mathbf{I}_M$ is the identity matrix of shape $M \times M$.

Algorithm 1: Kalman filter algorithm based on the state model in equation (8)

Initialize $\hat{\mathbf{e}}_0$, $\hat{\mathbf{P}}_0$, $i = 1$
while *True* **do**
  Prediction step:
  $\hat{\mathbf{e}}_{i|i-1} \leftarrow \mathbf{F}_i\hat{\mathbf{e}}_{i-1}$
  $\hat{\mathbf{P}}_{i|i-1} \leftarrow \mathbf{F}_i\hat{\mathbf{P}}_{i-1}\mathbf{F}_i^H + \mathbf{Q}_i$
  Estimation with autocalibration: get $\mathbf{y}_i$
  Update step:
  $\mathbf{S}_i \leftarrow \mathbf{D}_i\hat{\mathbf{P}}_{i|i-1}\mathbf{D}_i^H + \mathbf{R}_i$
  $\mathbf{K}_i \leftarrow \hat{\mathbf{P}}_{i|i-1}\mathbf{D}_i^H\mathbf{S}_i^{-1}$
  $\hat{\mathbf{e}}_i \leftarrow \hat{\mathbf{e}}_{i|i-1} + \mathbf{K}_i\left(\mathbf{y}_i - \mathbf{D}_i\hat{\mathbf{e}}_{i|i-1}\right)$
  $\hat{\mathbf{P}}_i \leftarrow \left(\mathbf{I}_M - \mathbf{K}_i\mathbf{D}_i\right)\hat{\mathbf{P}}_{i|i-1}$
  $i \leftarrow i + 1$
end

3.1 Discussion on the Evolution Matrix

The evolution matrix $\mathbf{F}_i$ is used to model the evolution of the vector $\mathbf{e}_i$ over time (7). The calibration vector $\mathbf{h}_i$, which can be controlled entirely, influences $\mathbf{e}_i$ predictably and can therefore be taken into account in $\mathbf{F}_i$. Conversely, the variations of $\mathbf{g}$ are unpredictable and thus have to be modeled by the process noise.

Firstly, if the calibration vector changes from $\mathbf{h}_{i-1}$ to $\mathbf{h}_i$, then the calibration error vector $\mathbf{e}_i$ at calibration step i becomes:

$$\begin{aligned}\mathbf{e}_i = \frac{\mathbf{g}}{\mathbf{h}_i} \iff \quad \mathbf{e}_i &= \frac{\mathbf{h}_{i-1}}{\mathbf{h}_i} \odot \frac{\mathbf{g}}{\mathbf{h}_{i-1}} \\ \iff \quad \mathbf{e}_i &= \text{diag}\left(\frac{\mathbf{h}_{i-1}}{\mathbf{h}_i}\right)\mathbf{e}_{i-1}\end{aligned} \tag{9}$$

with $\odot$ being the element-wise product operator and the divisions between the vectors being element-wise divisions. Thus, $\mathbf{F}_i = \text{diag}\left(\frac{\mathbf{h}_{i-1}}{\mathbf{h}_i}\right)$.

Secondly, if the estimate yielded by the Kalman filter is used to calibrate the radar, then:

$$\mathbf{h}_i = \frac{\hat{\mathbf{e}}_{i-1}}{\hat{e}_{i-1}[r]} \odot \mathbf{h}_{i-1} \tag{10}$$

Here, $\hat{e}_{i-1}[r]$ is the r^{th} element of $\hat{\mathbf{e}}_{i-1}$. The division by $\hat{e}_{i-1}[r]$ ensures that the reference element r of the array has a fixed complex gain. Then substituting Eq. 10 into Eq. 9 yields:

$$\mathbf{F}_i = \text{diag}\left(\frac{\hat{e}_{i-1}[r]}{\hat{\mathbf{e}}_{i-1}}\right) \tag{11}$$

In that case,

$$\hat{\mathbf{e}}_i = \mathbf{F}_i\hat{\mathbf{e}}_{i-1} = \begin{pmatrix}\hat{e}_{i-1}[r] & \hat{e}_{i-1}[r] & \dots & \hat{e}_{i-1}[r]\end{pmatrix}^T \tag{12}$$

In other words, this choice of $\mathbf{F}_i$ ensures that the complex gain response of each element of the array is $\hat{e}_{i-1}[r]$ to the best of the available knowledge.

Lastly, if the calibration vector does not change between two calibration steps, then $\mathbf{F}_i = \mathbf{I}_M$.

3.2 Discussion on the Measurement Matrix

In most cases, an autocalibration method yields an estimate of the calibration error $\mathbf{e}_i$ directly. In that case, $\mathbf{y}_i = \mathbf{e}_i + \mathbf{v}_i$, thus $\mathbf{D}_i = \mathbf{I}_M$.

However, there are some particular cases to consider. First, some calibration methods allow for the calibration of a subset of the array [4, 6, 7], which is desirable in some applications, such as reducing computation times on a very large array. In that case, $\mathbf{D}_i = \mathbf{J}$ with $\mathbf{J}$ a selection matrix of shape $S \times M$, with S the number of selected elements:

$$J[s,m] = \begin{cases} 1 & \text{if the } m^{\text{th}} \text{ element is the } s^{\text{th}} \text{ selected element} \\ 0 & \text{otherwise.} \end{cases} \tag{13}$$

For example, the matrix selecting elements 0, 2 and 5 from an array of 6 elements is:

$$\mathbf{J} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{14}$$

Furthermore, some calibration methods do not take into account the effect of coupling between elements (the matrix $\mathbf{C}$), which might be an issue in some cases. In the Kalman approach, the effect of coupling can sometimes be accounted for if $\mathbf{C}$ is known, thus compensating for this deficiency of the chosen autocalibration method. For instance, if the chosen autocalibration approach uses a single calibration target, then the actual estimated vector is not $\mathbf{g}$ but $\mathbf{g}'$:

$$\mathbf{g}' = \text{diag}\left(\mathbf{a}^*(\theta)\right) \text{diag}\left(\mathbf{g}\right) \mathbf{C}\mathbf{a}(\theta) \tag{15}$$

with $a(\theta)$ being the steering vector of the array for direction θ, which is the direction of the calibration target, and $(.)^*$ is the complex conjugate operation. diag $(\mathbf{a}^*(\theta))$ is here because the steering phase due to the direction of the calibration target is compensated for [6, 4, 7].

Then, some simple algebraic manipulations yield:

$$\mathbf{g}' = \text{diag}\left(\text{diag}\left(\mathbf{a}^*(\theta)\right) \mathbf{C}\mathbf{a}(\theta)\right) \mathbf{g} \tag{16}$$

It is worth noting that if $\mathbf{C} = \mathbf{I}_M$, then $\mathbf{g}' = \mathbf{g}$, as expected. Finally, this expression can be used in $\mathbf{D}$:

$$\mathbf{D} = \mathbf{\Omega}\text{diag}\left(\text{diag}\left(\mathbf{a}^*(\theta)\right) \mathbf{C}\mathbf{a}(\theta)\right) \tag{17}$$

with $\mathbf{\Omega}$ being $\mathbf{I}_M$ or $\mathbf{J}$ depending of the situation, as described previously.

3.3 Process Noise and Measurement Noise Matrices

The correlation matrices of process noise $\mathbf{Q}_i$ and measurement noise $\mathbf{R}_i$ should be chosen appropriately.

Ideally, $\mathbf{Q}_i$ should be chosen using known hardware data or statistical data. Alternatively, a good approximation could be $\mathbf{Q}_i = \sigma_{\mathbf{Q}}^2 \mathbf{I}_M$ with $\sigma_{\mathbf{Q}}$ a hyper-parameter to be determined.

Similarly, $\mathbf{R}_i$ should be chosen using statistical data of the calibration error of the chosen autocalibration method, or $\mathbf{R}_i = \sigma_{\mathbf{R}}^2 \mathbf{I}_M$ with $\sigma_{\mathbf{R}}$ another hyper-parameter to be determined.

4 Simulation Results

Simulations are done using Python (v3.13.2), as well as the libraries NumPy (v2.1.3) and Matplotlib (v3.10.1) for data array manipulations and data plots, respectively.

In the simulations, the complex gain $\mathbf{e}_i$ of an array of $M = 128$ elements is estimated using the autocalibration methods described in [6] with $P = 1$ target. These successive gain estimations are then fused using the method described in this article. The calibration error is then compared to the results of the autocalibration method without fusion (discard-and-replace method), which consists of discarding the previous calibration $\mathbf{h}_i$ and estimating the complex gain vector to find $\mathbf{h}_{i+1}$.

The signal-to-noise ratio (SNR) is approximately 10 dB after pulse compression which is quite high but necessary for the chosen autocalibration method. A lower SNR is not an obstacle as long as the chosen autocalibration method can function at the given SNR. For each radar echo, $N = 4096$ samples are taken.

The process noise and measurement noise matrices are chosen arbitrarily:

$$\mathbf{Q}_i = 0.1\mathbf{I}_M \quad \text{and} \quad \mathbf{R}_i = 2\mathbf{I}_S \tag{18}$$

In addition, the Kalman filter is initialized as follows:

$$\hat{\mathbf{P}}_0 = 10\mathbf{I}_M \quad \text{and} \quad \hat{\mathbf{e}}_0 = \begin{pmatrix} 1 & 1 & \dots & 1 \end{pmatrix}^T \tag{19}$$

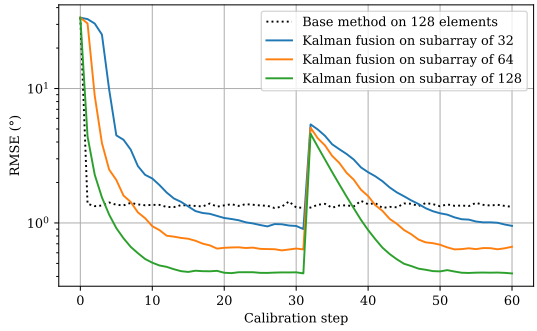
Here, $(.)^T$ is the transposition operator. This choice of noise matrices and initialization is not particularly efficient, and a better one could be found. However, it is sufficient as a proof of concept. Finally, the 0^{th} element of the array is chosen as a reference element and an arbitrary real coupling matrix is chosen, and used when specified:

$$\mathbf{C} = \begin{pmatrix} 1 & 0.7 & 0.5 & 0 & \cdots & 0 \\ 0.7 & 1 & 0.7 & \ddots & & \\ 0.5 & 0.7 & 1 & \ddots & & \\ \vdots & \ddots & \ddots & \ddots & & \\ 0 & & & & & 1 \end{pmatrix} \tag{20}$$

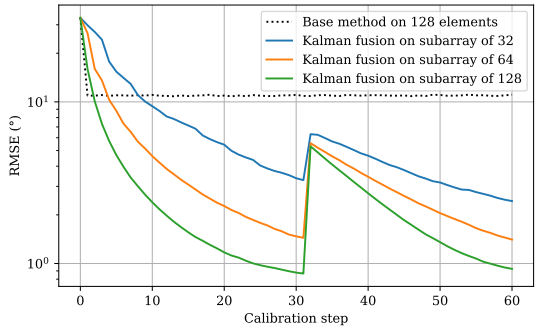
The simulation results are shown in Fig. 2. Only the phase error is displayed, since this error is usually the parameter of interest in an autocalibration problem. The vector $\mathbf{g}$ is changed slightly half-way through the simulation by adding an additional uniform phase to $\mathbf{g}$. This is done to show how the Kalman filter responds to such a changes in the target vector. The discard and replace method is not impacted by this change since it does not keep track of previous calibration vectors.

As shown in Fig. 2, the Kalman fusion approach yields a lower or equal Root Mean Square Error (RMSE) than the discard-and-replace method. In addition, it also yields a lower RMSE when there is coupling, which is expected since the base method does not take coupling into account (see Fig. 2b and Fig. 3).

Furthermore, the number of selected elements S (matrix $\mathbf{J}$) has an influence on the performance of the algorithm. A low number of selected elements yields a lower quality estimate ($S = 32$ in Fig. 2), while a high number of selected elements S yields a better estimate ($S = 128$ in Fig. 2), but it also increases the computation time for two reasons: the



(a) No coupling between element



(b) With coupling $\mathbf{C}$ between elements (see Eq. 19)

Figure 2: RMSE of the phase error as a function of the number of calibration steps for the discard and replace method and the Kalman fusion approach and different values of S. A Monte-Carlo averaging is done over 30 simulations.

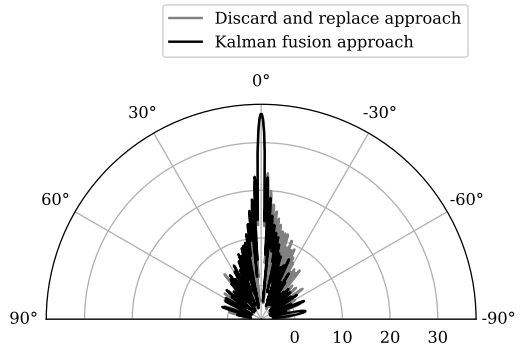


Figure 3: Radiation pattern (dB) of the array calibrated with a discard-and-replace approach (gray), and with the Kalman filter approach (black) for $S = 32$ selected elements with coupling.

autocalibration algorithm needs to estimate a higher number of parameters, and the computational cost of the Kalman filter's update step increases with the size of the measurement vector $\mathbf{y}_i$ non-linearly because of the computation of $\mathbf{S}_i^{-1}$.

The better estimate obtained with the Kalman fusion approach ensures lower side-lobe levels, and reduces the impact of coupling between elements (see Fig. 3).

5 Conclusion

The results of this study demonstrate that the proposed data fusion approach using a Kalman filter improves measurably the accuracy of autocalibration in antenna arrays by using the correlations in successive complex gain estimates. Additionally, this approach allows for the concurrent use of multiple calibration methods and supports the calibration of subsets of the array, making it particularly effective for large arrays and real-time applications. These findings highlight the potential of the Kalman filter-based technique to optimize radar system performance and ensure reliable operation for the entire lifetime of the radar at a low cost.

Acknowledgment

This work was funded by Thales LAS France under the ANRT CIFRE Defense program, with the support of the AID (*Agence Innovation Défense*) from DGA, under grant 2023643.

References

[1] I. Şeker, "Calibration methods for phased array radars," in *Proceedings Volume 8714, Radar Sensor Technology XVII* (K. I. Ranney and A. Doerry, eds.), (Baltimore, Maryland, USA), p. 87140W, May 2013.

[2] E. L. Shoshin, "External Calibration Methods for Radar Polarimeters," *Measurement Techniques*, vol. 64, pp. 228–237, June 2021.

[3] Kuan-Min Lee, Ruey-Shi Chu, and Sien-Chang Liu, "A built-in performance-monitoring/fault isolation and correction (PM/FIC) system for active phased-array antennas," *IEEE Transactions on Antennas and Propagation*, vol. 41, pp. 1530–1540, Nov. 1993.

[4] G. Babur, D. Caratelli, and A. Mirmanov, "PHASED ARRAY CALIBRATION BY BINARY COMPRESSED SENSING," *Progress In Electromagnetics Research M*, vol. 73, pp. 61–70, 2018.

[5] J. Geiss, E. Sippel, M. Hehn, and M. Vossiek, "Antenna Array Calibration Using a Sparse Scene," *IEEE Open Journal of Antennas and Propagation*, vol. 2, pp. 349–361, 2021.

[6] J. Jiang, F. Duan, J. Chen, Z. Chao, Z. Chang, and X. Hua, "Two New Estimation Algorithms for Sensor Gain and Phase Errors Based on Different Data Models," *IEEE Sensors Journal*, vol. 13, pp. 1921–1930, May 2013.

[7] K. S. Kim, E. Yang, and N. H. Myung, "Self-Calibration of an Active Uniform Linear Array Using Phase Gradient Characteristics," *IEEE Antennas and Wireless Propagation Letters*, vol. 18, pp. 497–501, Mar. 2019.

[8] S. Cao, Z. Ye, D. Xu, and X. Xu, "A Hadamard Product Based Method for DOA Estimation and Gain-Phase Error Calibration," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, pp. 1224–1233, 2013.

[9] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of the ASME–Journal of Basic Engineering*, vol. 82, no. Series D, pp. 35–45, 1960.