Recognizing human activity using deep learning with WiFi CSI and filtering

Sang-Chul Kim
Department of Computer Science
Kookmin University
Seoul, Rep of Korea
Sckim7@kookmin.ac.kr

Yong-Hwan Kim
Department of Computer Science
Kookmin University
Seoul, Rep of Korea
brightface@kookmin.ac.kr

Abstract: We are living in the era of the Internet of Things, where it is easy to find network access points (APs). APs could be useful for more than just connecting to the Internet. The presence of a human between two APs, as well as human behavior, causes a change in the waveform of a WiFi signal. In a previous research, we have explained how changes in waveforms affect the channel state information of the signal and how machine learning can utilize that information to recognize and predict human behavior. In this paper, we explain the limitation of the last paper and provide a solution for improving the limited performance, which is preprocessing. Kalman filtering improved the training accuracy by 2%. In conclusion, the overall Kalman filter is good for suppressing sudden signal errors such as those from hardware malfunctioning.

Keywords: LSTM; CSI; Kalman Filter; IoT; CNN; LoS; Smart Home; Smart Office

I. Introduction

Due to the recent developments in the Internet of Things technology, WiFi signals are captured by mobile devices whenever possible.

WiFi AP signals have many advantages and are useful in many areas. They result from network access points (Aps), including information such as channel state information (CSI). We use this information to communicate and develop many technologies. The CSI shows the distribution channel road of the signal from which it was transmitted and could be reflected or scattered based on the surrounding object [1].

Therefore, the problem is found in the hardware noise in the signal or the weak signal that occurs due to fading. The noise from the hardware is unavoidable using commercial NIC, which was originally not designed to measure the CSI. A commercial NIC with an open-source firmware inevitably contains noise, which can be filtered out using the Kalman filter. Extracting the features and distinguishing the difference between walking and standing have become clear. The feature you can find at each signal is distinguishable. However, the signal pattern difference between sitting and walking is unclear [2].

We used two Intel WiFi Link 5300 wireless NICs for transmission and receipt, with the receiver using three antennas to receive WiFi signals. If a person is located between a transmitting location and a receiving location, the transmitted WiFi signal is reflected differently. The previous signal is transmitted via a new path from the transmitting location. The CSI from the receiving antenna possesses its own characteristics in the following structure: $A \in \mathcal{C}^{N_{sa} \times N_{sc} \times N_{tx} \times N_{rx}}$, where N_{sa} and N_{sc} are the numbers of sampled WiFi packets and subcarriers, respectively. N_{tx} and N_{rx} are the numbers of transmitting and receiving antennas, respectively [3]. In our setting, $N_{sc} = 30$, $N_{tx} = 3$, and $N_{rx} = 3$.

We took the CSI at each sampling timestamp as a multichannel tensor,($a_i \in \mathcal{C}^{30\times3\times3}$, $i \in [1, N_{sa}]$). We can extract features from the data for machine learning. We used the Kalman filter and developed a model that enables us to recognize human behavior using the LSTM.

The recognition of human behavior using unfiltered CSI data had much noise. Our world includes many objects in various environments; thus, much noise occurs from lack of hardware liability. Model performance, which is necessary in the real world, is also lacking. To use data preprocessing and filtering, let the model ensure a better performance when the surrounding object changes.

Subsequently, Chapter 2 explains the results and limitations of the previous paper. The WiFi Channel State Information-based Human Activity Recognition and Prediction (reference) experiment under a controlled environment means that it cannot guarantee the accuracy under various situations. The performance that the previous model produced was high enough to be only

2021 International Conference on Artificial Intelligence in Information and Communication (ICAIIC) | 978-1-7281-7638-3/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICAIIC51459.2021.9415247

utilized under controlled situations. In other words, it cannot be used in practice. Chapter 3 presents the Kalman filters used and how they improved the performance accuracy. A comparison of the performance obtained in the previous paper and that after preprocessing herein is also performed. Finally, Chapter 4 discusses a technical example and the conclusions.

II. PREVIOUS RESULT AND LIMITATION

A previous research [4] proved that the deep learning methodology is applicable for human behavior recognition. Figure 1 shows a confusion matrix illustrating that the model accuracy was over 90%. The benefit of using deep learning over traditional machine learning is that deep learning can distinguish more varieties of activity and is stronger to change or the surrounding environment. However, using the LSTM without any preprocessing can also let the machine train noise or suppress it from learning features out of the dataset.

We tested the model in different circumstances other than that where we collected the dataset, which is an empty lab. In a room with a taller ceiling and a wider space, the signal dataset was not strong enough to extract the feature. Furthermore, the NIC used in this experiment did not produce a stable CSI because it was not originally made to collect the CSI.

The accuracy of the deep learning model can be improved in two ways: 1) increase the size of the training dataset; and 2) use filters to preprocess the dataset. This study focused on filtering datasets to minimize the noise and extract the feature to improve the performance.

Walk	Stand	Empty	Sit down
92.5 %	90 %	100 %	75 %

III. METHODOLOGY

3.1. Kalman Filter

We used three filters here. First is the Kalman filter [5], which is an algorithm that uses a series of measurements observed over time, contains statistical noise and other inaccuracies, and produces estimates of unknown variables that tend to be more accurate than those based on a single measurement alone by estimating a joint probability distribution over the variables for each timeframe.

This procedure is based on linear dynamical systems discretized in the time domain and modeled on a Markov chain built on linear operators perturbed by errors that may include the Gaussian noise. The system state is represented as a vector of real numbers. At each discrete time increment, a linear operator is applied to the state to generate a new state with some noise mixed in, and optionally, some information from the controls on the system, if they are known. Another linear operator mixed with more noise generates the observed outputs from the true ("hidden") state. The Kalman filter may be regarded as analogous to the hidden Markov model [6], with the key difference of the hidden state variables taking values in a continuous space. A strong analogy exists between the equations of the Kalman Filter and those of the hidden Markov model [7]. The Kalman filter model assumes that the true state at time k evolves from the state at $(k-1)$ according to the abstract state transition equation of the system:

$$x_k = F_k x_{k-1} + B_k u_k + w_k \quad (1)$$

In Formula (1), $F_k$is the state transition model applied to the previous state x_{k-1}.

The state transition model herein is the identity matrix. There is no factor to cause change in the empty space.

B_k is the control-input model applied to the control vector u_k. This part was omitted because the Kalman filter is only used to filter statistical and hardware noises. The external influence was not considered.

w_k is the process noise assumed to be drawn from a zero mean multivariate normal distribution, $\mathcal{N}(0, Q_k)$. At time k, an observation (or measurement) z_k of the true state x_k is made according to $z_k = H_k x_k + v_k$, where H_k is the observation model that maps the true state space into the observed space, and v_k is the observation noise assumed to be a zero mean Gaussian white noise with covariance R_k: $v_k \sim \mathcal{N}(0, R_k)$ The initial state and the noise vectors at each step $\{X_0 w_1 w_k v_1 v_k\}$ are all assumed to be mutually independent.

The Kalman filter in the discrete linear system model is

$$X_k = \Phi_{k,k-1} X_{k-1} + \Gamma_{k,k-1} W_k \quad (2)$$

$$Z_k = H_k X_k + V_k \quad (3)$$

$\Phi_{k,k-1}$ is a state deviation transition matrix (identity), and H_k is the measurement matrix (garbage value in between 300 and 0) representing the relationship between the amount of measurement and status [8]. $\Gamma_{k,k-1}$ is the system noise driving matrix. The statistical characteristics of the system noise W_k and the measurement noise V_k are as follows:

$$E[W_k] = 0, R_{ww}(k,j) = Q_k\delta_{kj} \quad (4)$$

$$E[V_k] = 0, R_{vv}(k,j) = R_k\delta_{kj} \quad (5)$$

$$R_{wv}(k,j) = 0 \quad (6)$$

Formulas (4) and (5) assume that the mean is a zero Gaussian distribution, and the covariance at this time is a normal distribution of Q and R. The noise is expressed through Q and R, respectively. Formula (6) explains that the process and measurement noises must be mutually exclusive (Q = 0.1, R = 0.01). Based on the discrete linear system model, the Kalman filter works as follows [9]:

Step 1: Calculation of the step prediction of the mean square error matrix.

$$P_{k,k-1} = \Phi_{k,k-1}P_{k-1}\Phi_{k,k-1}^T + \Gamma_{k,k-1}Q_{k-1}\Gamma_{k,k-1}^T \quad (8)$$

Step 2: Calculation of the state step prediction.

$$\hat{X}_{k,k-1} = \Phi_{k,k-1}X_{k-1} \quad (9)$$

Step 3: Calculation of the Kalman filter gain.

$$K_k = P_{k,k-1}H_K^T(H_kP_{k,k-1}H_K^T + R_k)^{-1} \quad (10)$$

$$X_k = \hat{X}_{k,k-1} + K_k(z_k\text{-H}\hat{X}_{k,k-1}) \quad (11)$$

Step 4: Calculation of the mean square error matrix of the filter.

$$P_k = (I - K_kH_k)P_{k,k-1} \quad (12)$$

Step 5: Calculation of the status filter value.

$$X_k + \Phi_{k,k-1}X_{k-1} + K_{k[r_k - h(\hat{X}_{k,k-1})]} \quad (13)$$

Steps 1 and 2 push the time from the k − 1 moment to the k moment; thus, they are collectively called the time update process (predict). Steps 3–5 use the observed quantity on time to update the value; therefore, they are called the measurement update process(correct).

First, we must distinguish between two terms that are similar, $P_{k,k-1}$ and P_{k-1}. The meaning of ^ on x is the "estimated value," called a posteriori state. $P_{k,k-1}$ and $\hat{X}_{k,k-1}$ mean "before," which is called a priori state.

In the "Predict" phase, the state of the next measurement time and the error covariance is predicted. The Kalman gain K is then calculated using the calculated error covariance value and the measurement noise covariance as in the first equation of the "Correct" step. Next, as in the second equation of the "Correct" step, the difference between the value z_k obtained after the actual measurement and the predicted measured state value H * $\hat{X}_{k,k-1}$ is multiplied by the Kalman gain K to calculate the previously predicted state value $\hat{X}_k$. Finally, the measurement is made as in the third equation of the "Correct" step; thus, it also updates the error covariance value.

Looking first at the second term in the second equation of the "Correct" step, this is the correction term with the name "measurement innovation." This indicates how much correction should be applied to the previous system state prediction by the newly measured data. In this case, if you look closely at the first equation of the "Correct" step, if the sensor noise is large, that is, if the measurement noise covariance is large, the K value of the Kalman gain decreases. Consequently, the confidence in the data to be measured will be lowered, such that the system state prediction is weighted rather than the measured value. Conversely, when the sensor noise is small, that is, when the measurement noise covariance is small, the Kalman gain K becomes large, which means that the new data to be measured is reliable.

In other words, when calculating the Kalman gain K, we use the variances of the process and measurement noises. If the process noise is more dispersed than the measurement noise, the measurement data will be multiplied by a larger weight and updated also by being multiplied by a larger weight.

The (n * n) matrix $\Phi_{k,k-1}$ and the (n * l) matrix $\Gamma_{k,k-1}$ can be viewed as systematic linear system equations that compute the system state at k, which is the current time from the system state at the previous time k − 1. The (m * n) matrix H is the value associated with the actual measurement data z_k. In the Kalman filter equation, the $\Phi_{k,k-1}$, $\Gamma_{k,k-1}$, and H matrices are system-dependent values that do not change. Moreover, H is determined by how the measurement is made. The Kalman filter must have unchanged matrices $\Phi_{k,k-1}$, $\Gamma_{k,k-1}$, H, Q, and R, and we assume that there are initial values $\hat{X}_{k-1}$ and P_{k-1} before the "predict" and "correct" routines begin. We then obtain the "predict" routine value with the unchanged matrix and initial values. Using this value, the next predicted value is corrected using the Kalman gain K and the measured value z_k. The error covariance value is also corrected. This process is repeated using a Kalman filter.

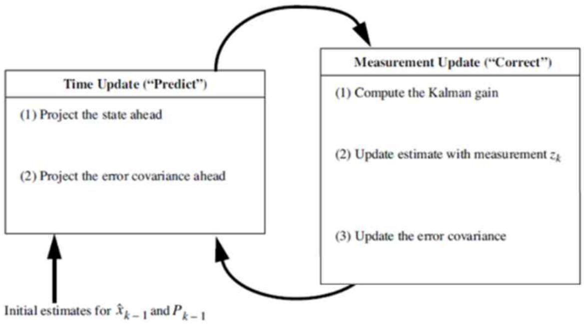
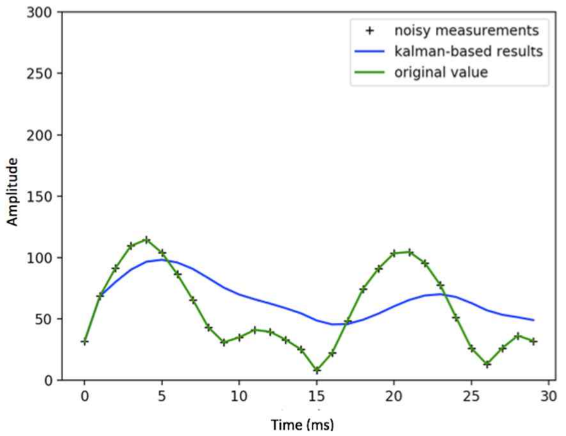


Figure 2 shows the original value of the CSI amplitude and that filtered with the Kalman filter.
The Kalman filter can filter both statistical noise and other inaccuracies.



	default	*Kalman filtered*
Accuracy	90	91.5

Figure 2

IV. Conclusion

Kalman filter is a recursive algorithm for tracing back the signal and filtering off-trend data. This algorithm is adequate for dealing with sudden signal changes (e.g., noise from the hardware level error or from more than one person standing in between two NICs).

Acknowledgments

"This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the ITRC(Information Technology Research Center) support program(IITP-2021-0-01396) supervised by the IITP(Institute for Information & Communications Technology Planning & Evaluation)"

References

[1] Ali K., L. A, "Keystroke Recognition Using WiFi Signals", ACM MobiCom, 2015.

[2] Anguirre S, H. S, "Two novel automatic frequency tracking loops", IEEE Transactions on Aerospace and Electronic Systems, Vol. 25, Iss. 5 1989.

[3] Chun-xiang WU, H. S, "A Wireless Signal Denoising Model for Human Activity Recognition", AICS, 2016.

[4] Kim, T. "WiFi Channel State Information-based Human Activity Recognition with machine learning, " ICAIIC, 2018.

[5] Yandu LIU, D. Y. "Signal Frequency Estimation Based on Kalman Filtering Method", Equipment Academy. Beijing, 2016

[6] Hamilton, J. "The Kalman Filter", Princeton University Press, Chapter 13, 1994.

[7] Roweis, S., & Ghahramani, Z. "A unifying review of linear gaussian models", Neural Computation, Vol. 11, Iss. 2 , 1999

[8] Jiguang Lv, D. M. "Robust Device-Free Intrusion Detection Using Physical Layer Information ofWiFi Signals", MDPI, 2019.

[9] Wang W., L. A. "Understanding and Modeling of WiFi Signal Based Human Activity Recognition", ACM MobiCom, 2015