

Received July 18, 2020, accepted August 4, 2020, date of publication August 13, 2020, date of current version September 14, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3016277

# ANN Based Learning to Kalman Filter Algorithm for Indoor Environment Prediction in Smart Greenhouse

ISRAR ULLAH<sup>1,3</sup>, MUHAMMAD FAYAZ<sup>2</sup>, NASIR NAVEED<sup>3</sup>, AND DOHYEUN KIM<sup>1</sup>

<sup>1</sup>Computer Engineering Department, Jeju National University, Jeju City 63243, South Korea

<sup>2</sup>Department of Computer Science, University of Central Asia, Naryn 722918, Kyrgyzstan

<sup>3</sup>Department of Computer Science, Virtual University of Pakistan, Lahore 44000, Pakistan

Corresponding author: Dohyeun Kim (kimdh@jejunu.ac.kr)

This work was supported in part by the Energy Cloud Research and Development Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT (MSIT) under Grant 2019M3F2A1073387, and in part by the Institute for Information and Communications Technology Planning and Evaluation (IITP) Grant funded by the Korean Government, MSIT (AutoMaTa: Autonomous Management Framework Based on Artificial Intelligent Technology for Adaptive and Disposable IoT) under Grant 2018-0-01456.

**ABSTRACT** In this article, we have proposed a learning to prediction based novel approach for improving the accuracy of prediction algorithms in dynamic conditions. The proposed model is composed of two modules, including the prediction module and the learning module. The learning module is responsible to regularly examine the prediction module and tune its performance by assessing its outcomes together with any other external parameters that can affect its performance. In order to determine the effectiveness of the proposed idea, a learning module based on the artificial neural network (ANN) is developed for improving the accuracy of the Kalman filter algorithm. Experimental investigations are conducted in a greenhouse indoor environment to accurately predict indoor climate parameters (temperature,  $CO_2$ , and humidity) from noisy sensors readings using the Kalman filter algorithm. Among the various components of the Kalman filter algorithm includes a fixed value of  $R$  (observation error covariance), which significantly degrades the performance of the Kalman filter algorithm in dynamic conditions. Greenhouse sensor readings are affected by changing external environmental conditions and internal greenhouse actuators operations. The amount of error in current readings is estimated using ANN-based learning modules to update the parameter  $R$  in the corresponding Kalman filter module. Performance evaluation of the proposed model based on learning is conducted in two different case studies using average and maximum based error models. Experimental results show that the prediction accuracy of the conventional Kalman filter is significantly improved by proposed learning to prediction scheme.

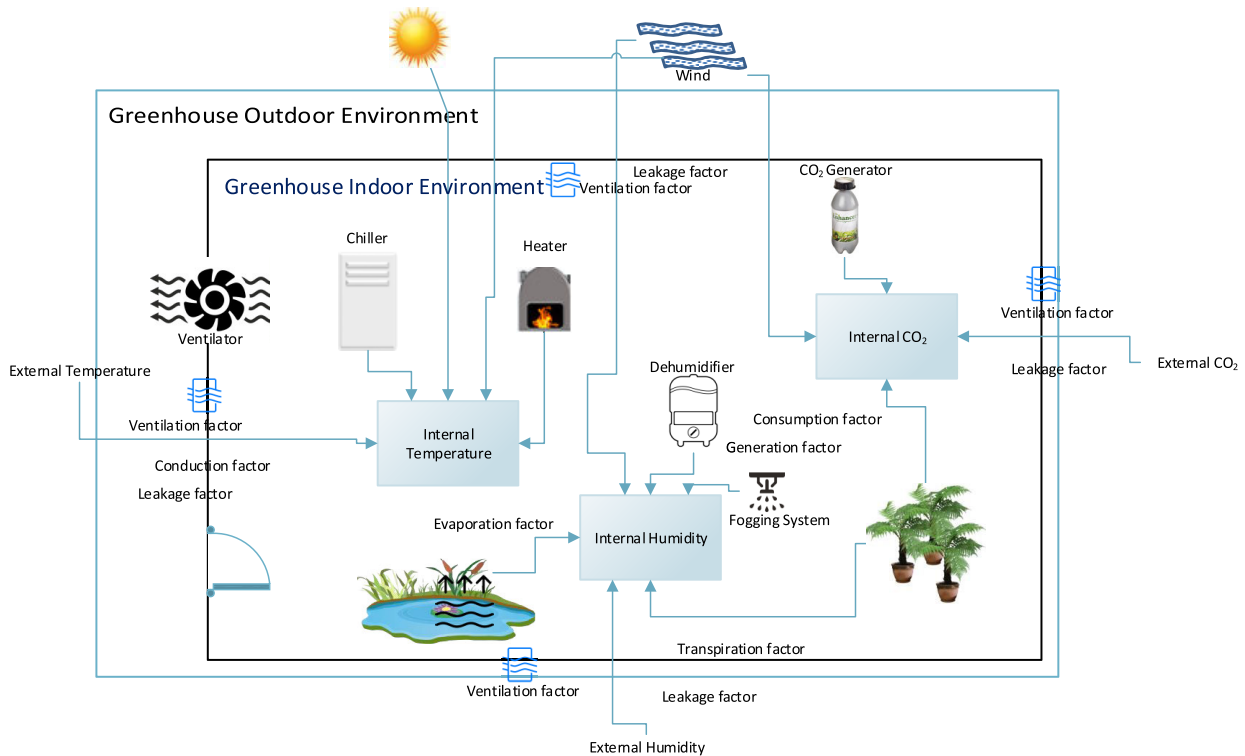
**INDEX TERMS** Artificial neural networks (ANN), Kalman filter, learning to prediction, smart greenhouse.

## I. INTRODUCTION

Due to global changes in climate conditions, conventional farming faces a lot of challenges, such as heavy rainfall, storms, inadequate freshwater, etc. It is estimated that the population of the world would be approximately 9.7 billion in 2050 [1]. Hence the FAO urges and encourages the usage of modern tools and technologies in all kinds of farming to achieve the desired target of food production for the future world. According to an estimate, worldwide food production

must rise by 50% [2]. Currently, 821 Million people suffer under nutrition [3], and it will rise with an increase in population if necessary corrective measures are not taken. Therefore, high importance is given by UN to achieve the Zero hunger (Sustainable Development Goals (SDG)-2) by 2030 [4]. Smart farming is considered as a key solution to fulfill the food requirement of the growing population. Just like fish production is supported by Aquaculture, similarly agriculture production can be improved with Greenhouses that have the potential of achieving 10-12 times higher production than open air cultivation. Greenhouse is a framed structure with transparent covering, providing partially or fully

The associate editor coordinating the review of this manuscript and approving it for publication was Jonghoon Kim<sup>1</sup>.



**FIGURE 1.** Typical model of Greenhouse environment.

controlled environment for optimum productivity, throughout the year. Figure 1 shows a typical model of greenhouse with essential components. Efficient operation and management of Greenhouse requires a careful and sound understanding of various associated greenhouses processes including photosynthesis, transpiration, relative humidity in the environment, respiration, vegetative and generative plant growth, etc.

The Greenhouse industry has received significant attention and experienced tremendous growth in recent past across the globe. Greenhouse provides a year-round production facility for fresh vegetables with around 50% increased production rate in comparison to open-air cultivation. Contextual information about greenhouse indoor environments such as temperature, humidity, illumination, etc. can be collected using IoT devices. From current context information, we can predict future conditions and resources requirements using different machine learning algorithms. It is extremely important to predict energy needs in the future for individuals as well as for power generating companies. In order to reduce the energy expenses, users can choose alternate ways, i.e. solar system etc. when feasible. Knowledge about energy requirement in advance helps in optimization of the operational hours and resources. Furthermore, appropriate capacity planning can be carried out to utilize renewable energy resources and hence helps to achieve the objective of the zero-energy environment. It is also very useful for power companies to manage energy production and accurately distribute power loads. Korean Energy Economics Institute (KEEI) South Korea, stated in

a report that residential and commercial sectors consumes 40% of total energy. Greenhouses are no exception, energy consumption and labor cost in greenhouses accounts for more than 50% of the cost of greenhouse production. Thus, a minor improvement in performance can lead to significant cost reduction. Electricity is a major factor in the overall expenses of greenhouse, hence it is extremely necessary to utilize the energy efficiently. Therefore, the development of an intelligent IoT-based solution to accurately predict greenhouse indoor climate conditions is highly desirable for achieving energy efficiency and maximizing plant productivity.

The latest advancement in communication, computation and machine learning technologies have assisted in uplifting the living standard in one way or another. Machine learning algorithms are based on knowledge and complex patterns that are usually extracted from current and historical data to make informed decisions regarding the future for maximization of profit and minimization of losses [5]. These forecasting algorithms are usually trained using some historical data. Afterwards, these trained models are used in the designated applications environment. As these algorithms are trained for a particular environment, therefore their performance gets degraded when operational conditions are changing.

This study is focused on enabling prediction algorithms to adapt with changing environmental conditions dramatically. Previously, we have developed a general architecture to improve the performance of the prediction algorithm using the learning module [6]. In this article, we have used the

same model in a smart greenhouse environment to predict indoor parameters accurately. Greenhouse indoor environment is continuously changing due to the operation of various actuators installed inside the greenhouse. Furthermore, external weather conditions also have a strong influence on the greenhouse indoor climate. The conventional Kalman filter algorithm fails to predict actual parameter in such a dynamic conditions. Therefore, we propose an improved learning to prediction algorithm to improve the performance of Kalman filter algorithm. The learning module is based on ANN and it continuously monitors the performance of prediction algorithm by analyzing its output. Learning module also considers other internal and external factors that have influence on the performance of prediction algorithm. Internal factors includes operational status of various actuators inside greenhouse and external factors include weather conditions such as wind speed, solar radiation, etc. After analysis, ANN based learning module updates the parameter  $R$  (observation error covariance) to improve the prediction accuracy of Kalman filter algorithm. In comparison to existing related studies, the key contributions of this study are highlighted as follows.

- A general conceptual model for learning to prediction is presented for improving the accuracy of prediction algorithms in dynamic conditions.
- Proposed model evaluation and experimental analysis is conducted in Greenhouse environment to accurately predict indoor climate conditions from noisy sensor readings.
- ANN algorithm is used to improve the performance of the conventional Kalman filter algorithm in dynamically changing external conditions.

The Remainder of the paper is structured as follows: Section II presents brief overview of related studies. Working of Kalman Filter algorithm for indoor climate prediction is explained in Section III. Conceptual design of the proposed model along with description of the selected case study is given in Section IV. Section V is dedicated to present experimental setup, noise models, simulation scenarios, and training/testing of learning modules. Comprehensive performance analysis of proposed model is presented in Section VI. Finally, this study is concluded in Section VII.

## II. RELATED WORK

Numerous researchers have developed many methods for controlling greenhouse environment. Akkas *et al.* proposed a methodology based on IoT in order to control the greenhouse temperature, light pressure, and humidity [7]. Yongtao *et al.* suggested an approach based on the prediction model in order to fulfill the energy demand. In this method the unknown factors of the model are tuned through particle swarm optimization algorithm [8]. Chen *et al.* proposed another model based on HPSO-GA in order to predict energy demand in smart greenhouse [9]. Similarly, another model has been developed based on ANN to predict energy in greenhouse [10]. However, these studies do not consider the impact of external weather conditions and resultant system dynamics.

Machine learning algorithms are extensively used in a lot of applications, such as in stock market prediction, risk prediction, weather prediction, etc. A lot of prediction algorithms have been proposed, such as Kth-nearest neighbor (KNN), artificial neural network (ANN), classification and regression tree (CART), etc. KNN is a simple and a very important algorithm that is used normally for both regression and classification [11], [12]. Support vector machines (SVM) are another important class of predictions algorithm. Normally, knowledge interpretation and extraction from a trained prediction algorithm by using training data in a human readable format are extremely hard. In such a situation, the decision tree has the capability to solve this problem. There are many types of decision tree algorithms among them the most significant algorithm is decision and classification tree (CART) [13], iterative dichotomizer 3 (ID3) [14], C4.5 [15], and chi-squared automatic interaction detector (CHAID) [16]. Recently, the CART algorithm has attracted a lot of researcher's attention and has been extensively used in different fields for prediction purposes. It is extremely difficult to solve the complicated interrelation between predictive parameters, in order to overcome this problem the CART has been introduced which has the ability to construct predictive models from the data. The construction of the CART model is normally carried by using two trees (classification and regression). The classification tree is designed for the dependent variable with order and the regression tree is designed for dependent variable without order. The error is the square of the difference between predicted and original values. In a random forest algorithm, numerous decision trees are generated by using random sampling [17], hence the overfitting problem is eliminated. However, the major problem with all these prediction algorithms is that once the model is trained using some historical data, it remains fixed and can be used only in the designated application environment. The main issue with these algorithms is their static nature and those are not suitable for a dynamic environment.

ANN is thought to be the most efficient natural-inspired prediction algorithm that mimic the working neurons in the human brain [18]. ANN algorithms can be applied to solve a lot of problems, i.e. regression, classification, clustering, pattern recognition, forecasting and time series data processing, etc. [19]–[22]. The ANN architecture is formed by using different layers (input, hidden, and output). There are different methods to train the ANN, such as error back propagation, gradient calculation methods, etc. [23]. In the conventional ANN methods, it is required to pre-process the inputs, extract features of interest and then normalization the extracted features. The number of neurons in the inputs layer depends on the number of inputs, the number of neurons in the hidden layer are decided through hit and trial method, the number of neurons in the output layer depends on the number of outputs. The selection of activation function is also decided through hit and trial method. Deep learning algorithms are an advance form of conventional ANN and are extensively used in different applications for predictions. There are

different types of deep neural networks, such as a convolutional neural network (CNN), recurrent neural network (RNN), long short-term memory (LSTM) [24]–[27] etc. In deep learning algorithms, the features extraction and reductions steps are eliminated, however their application in dynamic conditions is restricted due to the fixed trained model.

Many research studies are conducted to improve the performance of prediction algorithms. A commonly used approach is to use several prediction algorithms in combination to get better accuracy such as ensemble models [28], stacked generalization [29], [30]. Jacobs *et al.* proposed another method called mixture-of-experts with the aim that by combining the statistical estimations methods, the accuracy is increased as compared to a single estimation method [31]. Qiao *et al.* proposed a methodology based on an optimization algorithm to improve the prediction accuracy of short-term natural gas consumption [32], [33]. Several hybrid models are developed to improve the prediction accuracy e.g. integration of an improved whale swarm algorithm (IWOA) and empirical and relevance vector machine (RVM) [34], a hybrid model of wavelet transform (WT), stacked auto-encoder (SAE) and long short-term memory (LSTM) [35]. Another important family of predictive algorithms is reinforcement learning which enables the algorithm to improve its performance by using previous outcomes [36], [37]. These algorithms to constantly improves their performance and consequently gets converged. After convergence, the learning process is stopped and no change occurs with changes in the environment and thus fails to perform well in dynamically changing environment.

Coping with dynamically changing external conditions is a very challenging problem and has attracted tremendous research attention, particularly in the field of feedback control systems. The Presence of an external uncertain disturbance often keeps the system in an unstable state, often results in oscillation around the equilibrium position. The sliding mode control (SMC) algorithm is commonly used to achieve stability but it often results in chattering and high-speed switching. This can potentially damage the system if switching gain is not selected appropriately. Chengxiang Liu *et al.* have studied this problem and proposed an ANN-based adaptive solution (NNSMC) to address chattering and high-speed switching of control input problems in the conventional SMC for an uncertain robotic system [38]. The proposed system results in simplification of control design without requiring modeling of the complex dynamics, instead ANN module is designed to select optimal switching gain to improve system robustness and stability analysis is conducted using Lyapunov method dynamically. Dead-zone nonlinearities and output constraints have a significant impact on the performance of flexible structures and components. Boundary control schemes are commonly used for suppression of vibration and achieve stability. Zhijia Zhao *et al.* have developed an adaptive neural network based boundary control scheme to improve the stability of string systems while addressing the challenges

of dead-zone nonlinearities, output constraints and system uncertainties [39], [40]. They have used radial basis function neural network (RBFNN - a variant of ANN) is used to predict the unknown system information such as error estimate, a mass of the payload, the tension of the string, and external disturbance and robust control theory is employed to design disturbance adaption law for appropriate handling unknown external disturbance. The theme of these studies is similar as ours i.e., using ANN based learning module to improve the performance of conventional algorithm/schemes. Whereas, in this study, we have used the ANN algorithm to improve the performance of conventional Kalman filter algorithm in a dynamically changing external conditions. Furthermore, our field of application is also different.

The prediction algorithms must adapt to the changing conditions of the environment. This will require a method that can somehow - in one way or another - detect the changes in the environment. Consequently, the prediction algorithms can be adapted accordingly to avoid the degradation of performance. Usually, this is achieved by attaching a learning module with the prediction algorithm to tune its performance. For instance, Kang *et al.* proposed a method based on fuzzy inference-based for tuning the performance of the Kalman filter algorithm [41]. By using this model, the attitude of a humanoid robot is accurately estimated. Kalman filter smooths the noise of gyros sensors reading in order to predict the correct orientation of the robot. Nevertheless, when robot starts to move then the noise in gyro sensor occur. In order to overcome this problem, the accelerometers sensors have been used to detect the current state of the robot and fuzzy inference system has been used to tune the Kalman filter parameters. In the same way, Ibarra *et al.* suggested an approach named adaptive neuro-fuzzy inference for tuning Kalman filter algorithm for correct attitude estimation base on the gyroscope and acceleration sensors [42]. Markov models have also been used for this purpose [43]. Our proposed conceptual model for learning to prediction is inspired from these studies. However, in this study, we have used different learning and prediction algorithms with different sets of parameters in a Greenhouse environment to accurately predict indoor climate conditions from noisy sensor readings.

### III. KALMAN FILTER ALGORITHM FOR INDOOR CLIMATE PREDICTION

Kalman filter is a lightweight technique that is used to predict the actual state by using the previous state information without requiring all historical data. One of the most important components of the Kalman filter is Kalman gain commonly expressed as  $K$ . The value of the Kalman gain is updated when needed to control weights given to the system own predicted state or sensor reading. The working mechanism of the Kalman filter is illustrated in Figure 2.

Noise factors are often dependent on environmental conditions and can extremely affect the reading of the sensors. Three important factors namely temperature,  $CO_2$  level, and humidity, have been considered in the proposed work.

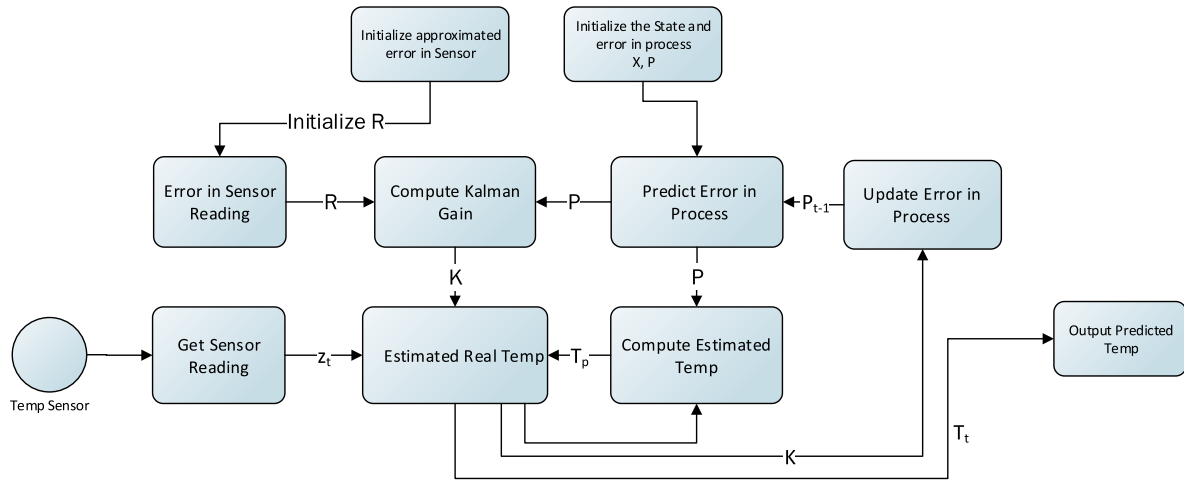


FIGURE 2. Working of Kalman filter algorithm for indoor temperature prediction in Greenhouse.

Next, the Kalman filter formulation has been presented to predict the actual temperature from noisy readings provided by the temperature sensor. The formulation for  $CO_2$  and humidity sensors readings is also the same and skipped here for brevity.

Let's suppose  $T_t$  is the greenhouse indoor temperature at time  $t$ . The Kalman filter algorithm makes an intelligent estimate about actual temperature  $T_{t+1}$  at time  $t + 1$  using internal prediction and current sensor readings. Firstly, predicted temperature  $T_p$  obtained from the last estimated value of temperature  $T_{t-1}$  using the following equation.

$$T_p = A \cdot T_{t-1} + B \cdot u_t \tag{1}$$

where  $T_p$  is Kalman filter internal predicted temperature,  $A$  represent the state transition matrix,  $B$  represents the control matrix and control vector is expressed as  $u_t$ .

We know that Kalman filter internal prediction will have some error which is determined by computing the covariance factor  $P_{predicted}$  as below.

$$P_{predicted} = A \cdot P_{t-1} \cdot A^T + Q \tag{2}$$

where  $A$  represents the state transition matrix and  $A^T$  is its transpose.  $P_{t-1}$  expresses the last value of the covariance factor and  $Q$  is the estimated error in the process.

Next, the updated value of Kalman's gain  $K$  is computed using an internal estimate about indoor temperature and updated covariance as below.

$$K = \frac{P_{predicted} \cdot H^T}{H \cdot P_{predicted} \cdot H^T + R} \tag{3}$$

where  $H$  represents the observation matrix,  $H^T$  is its transpose and  $R$  is the estimated error in the measurements process.

Let  $z_t$  be the recent reading of noisy temperature sensor at time  $t$ . Then, the following equation is used by Kalman's filter

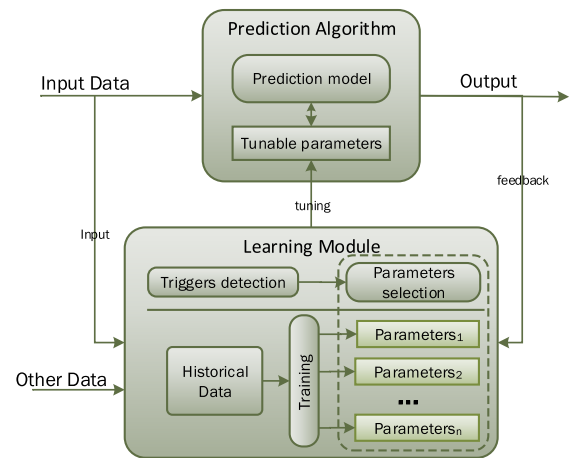


FIGURE 3. Conceptual model of proposed learning to prediction approach.

algorithm to predict the actual temperature of the greenhouse indoor environment.

$$T_t = T_{p_{predicted}} + K(z_t - H \cdot T_{p_{predicted}}) \tag{4}$$

Lastly, covariance factor  $P_t$  is updated using the following formula for the next iteration.

$$P_t = (I - K \cdot H)P_{predicted} \tag{5}$$

#### IV. PROPOSED LEARNING TO PREDICTION SCHEME

Prediction algorithms are very helpful in maximizing the benefits or avoiding potential losses by making informed decisions about an unknown future through learning from previous data. For that purpose, prediction algorithms usually rely on a training model that is learned from previous data



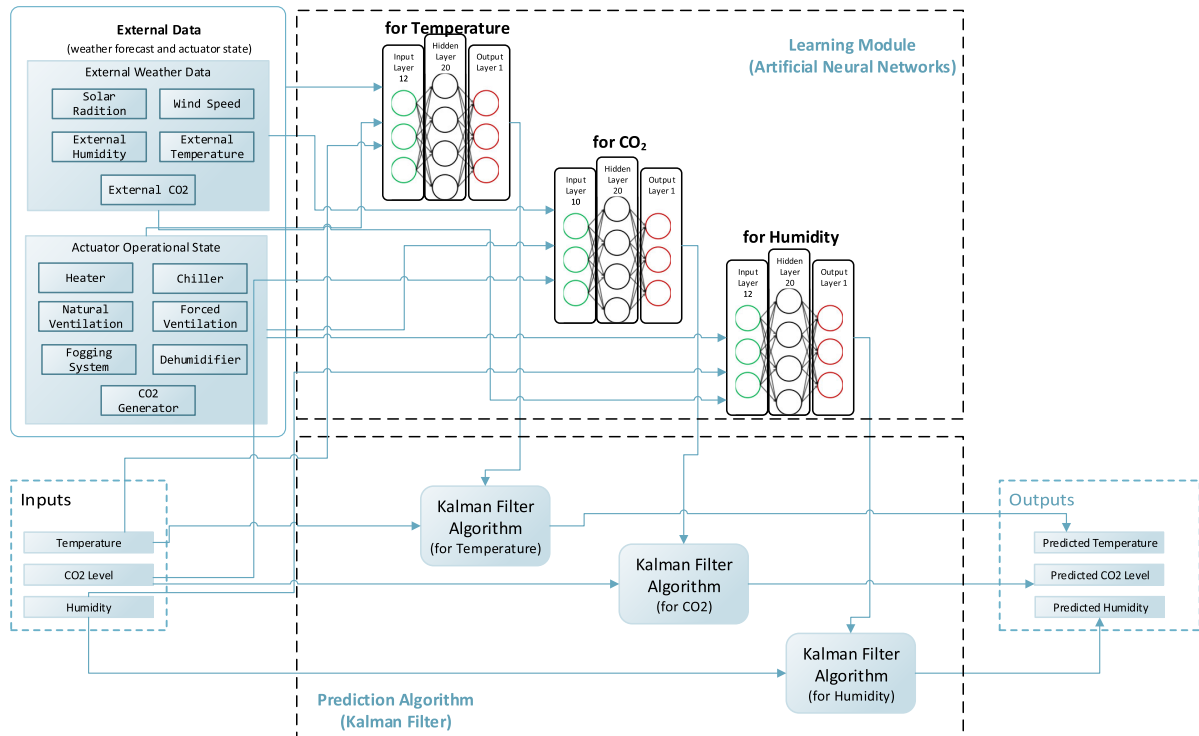


FIGURE 4. Abstract design of proposed ANN based learning to Kalman Filter algorithm.

before deployment. Such a prediction algorithm works fine when conditions remain the same as observed in training data. However, the major problem with this approach is that it restricts the capability of prediction algorithms to work in dynamic situations. In this study, we have proposed a novel learning to prediction based approach for improving the accuracy of prediction algorithms in the dynamic conditions. The proposed model is comprised of two modules, namely the prediction module and learning module. The learning module has the responsibility to regularly examine the prediction module and enhance the efficiency of the prediction module by assessing its outcomes together with any other external parameters that can affect the performance of the prediction module.

This study is concentrated on enabling prediction algorithms to adopt with changing environmental conditions dynamically. Previously, we have developed a general architecture to improve the performance of the prediction algorithm using the learning module as shown in Figure 3 [6]. In this article, we have used the same model in a smart greenhouse environment to predict indoor parameters accurately. The Greenhouse indoor environment is continuously changing due to the operation of various actuators installed inside the greenhouse. Furthermore, external weather conditions also have a strong influence on the greenhouse indoor climate. The Conventional Kalman filter algorithm [44], [45] fails to predict actual parameters in such dynamic conditions.

Therefore, we propose improved learning to prediction algorithm for performance improvement of the Kalman filter algorithm. The learning module is based on ANN and it continuously monitors the performance of the prediction algorithm by analyzing its output. The learning module also considers other internal and external factors that affects the performance of prediction algorithm. Internal factors include operational status of various actuators inside greenhouse and external factors include weather conditions such as wind speed, solar radiation, etc. After analysis, ANN based learning module updates the parameter  $R$  (which is further used to compute the Kalman gain  $K$ ) to improve the prediction accuracy of the Kalman filter algorithm.

Figure 2 presents the flow diagram of the Kalman filter that works perfectly fine when environmental conditions remains same. When error in sensor reading is changing because of some external factor, then updating of estimated error in the measurements ( $R$ ) is needed. In this work, the Greenhouse indoor scenario is considered where actuator operational conditions effect the sensors readings. The prediction accuracy of conventional Kalman filter algorithm suffer significant degradation under dynamic conditions. Figure 4 presents the abstract diagram of the proposed learning to prediction scheme. As we have three parameters (temperature, CO<sub>2</sub>, humidity), therefore three separate instances of Kalman filter algorithm are used. Each Kalman filter algorithm have its own learning module. Internal structure of these learning

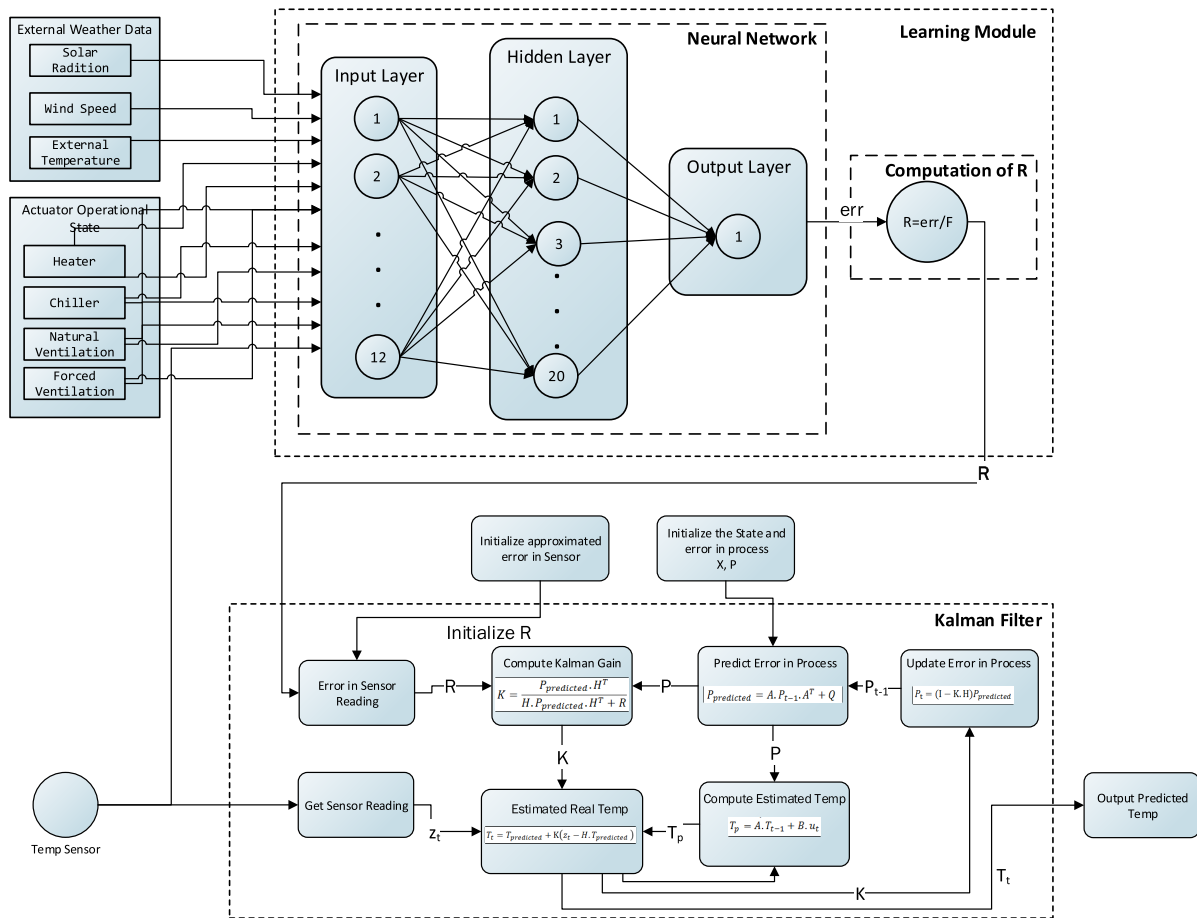


FIGURE 5. Detailed diagram for predicting greenhouse indoor temperature using proposed learning to prediction scheme.

modules is different as each module takes different set of input data.

Detailed design of the proposed learning to prediction model for temperature sensor readings is given in Figure 5. The learning module is based on ANN algorithm taking twelve (12) inputs i.e. 3 external environment parameters, 08 inputs related to selected actuators operation level and duration and current noisy temperature sensor readings. (see Table 1 for details). Experiments are conducted with varying number of neurons in hidden layer, different activation functions, learning algorithms and learning rates. The best results were obtained and are reported in this article with 20 neurons in hidden layer, sigmoid activation function, Levenberg-Marquardt learning algorithms and learning rate of 0.1. The prediction error in sensor reading is the generated outcome of the ANN. The predicted error is then divided by a static factor  $F$  for computing the error estimation in sensor reading, i.e.  $R$ . The updated value of  $R$  is then used as input to the Kalman filter for tuning its prediction accuracy through appropriate adjustment of Kalman gain  $K$ . The suggested learning to prediction approach makes able the Kalman filter

to correctly do the estimation of real temperature from noisy sensor reading with dynamic error rate.

Table 1 present the configuration summary of learning modules for temperature,  $CO_2$  and humidity sensing data.

## V. METHODS

### A. EXPERIMENTAL SETUP

Performance evaluation of proposed learning based prediction scheme is carried out through a custom-built greenhouse simulator by modeling greenhouse indoor environmental processes which take into account the impact of external environmental parameters and actuator operational level on indoor parameters. These applications are developed in Visual Studio C#. Table 2 present specification of the system and tools used in the development of these applications.

Accord.NET framework [46] is used for development of ANN based learning module. Configuration details of ANN algorithms for each parameter is given in Table 1.

For experimental analysis, we have used 15 days weather data obtained from online weather site Meteoblue [47] for

**TABLE 1. Configuration summary of ANN based learning modules for Greenhouse indoor parameters prediction.**

S. No.	Learning Module	Inputs	Hidden Layer	Output
1	ANN based Learning Module for tuning Kalman filter of Temperature Sensor Readings	1. External Temperature 2. Solar radiation 3. Wind Speed 4. Temperature Sensor Readings 5. Natural Ventilation Level 6. Natural Ventilation Duration 7. Forced Ventilation Level 8. Forced Ventilation Duration 9. Heater Level 10. Heater Duration 11. Chiller Level 12. Chiller Duration	Number of Neurons=20, Activation Function= Sigmoid ( $\alpha = 2$ ), Learning algorithm= Levenberg-Marquardt Learning rate=0.1	Predicted error is temperature readings (err) which is divided by a constant factor F to get updated value of R for corresponding Kalman filter algorithm.
2	ANN based Learning Module for tuning Kalman filter of CO <sub>2</sub> Sensor Readings	1. External CO <sub>2</sub> 2. Solar radiation 3. Wind Speed 4. Sensor Readings 5. Natural Ventilation Level 6. Natural Ventilation Duration 7. Forced Ventilation Level 8. Forced Ventilation Duration 9. CO <sub>2</sub> Generator Level 10. CO <sub>2</sub> Generator Duration	Number of Neurons=20, Activation Function= Sigmoid ( $\alpha = 2$ ), Learning algorithm= Levenberg-Marquardt Learning rate=0.1	Predicted error is CO <sub>2</sub> readings (err) which is divided by a constant factor F to get updated value of R for corresponding Kalman filter algorithm.
3	ANN based Learning Module for tuning Kalman filter of Humidity Sensor Readings	1. External Humidity 2. Solar radiation 3. Wind Speed 4. Humidity Sensor Readings 5. Natural Ventilation Level 6. Natural Ventilation Duration 7. Forced Ventilation Level 8. Forced Ventilation Duration 9. FogSys Level 10. FodSys Duration 11. Dehumidifier Level 12. Dehumidifier Duration	Number of Neurons=20, Activation Function= Sigmoid ( $\alpha = 2$ ), Learning algorithm= Levenberg-Marquardt Learning rate=0.1	Predicted error is Humidity readings (err) which is divided by a constant factor F to get updated value of R for corresponding Kalman filter algorithm.

**TABLE 2. System configuration for simulator implementation and performance analysis.**

Module	Specifications
CPU	Intel® Core(TM) i3-2120 CPU@ 3.30GHz, 3.30GHz
RAM	16 GB
Graphics	NVIDIA GeForce 9600 GT
Operating System	Window 7 Ultimate
Development Environment	Visual Studio Enterprise 2015 (C#) with .Net Framework 4.7
Libraries and tools	Accord.Neuro

Jeju, South Korea. The data includes outdoor temperature, humidity, wind speed and solar radiation information collected over hourly interval bases. In this study, our objective is to maintain Greenhouse indoor temperature, CO<sub>2</sub> level and humidity within desired user settings. Original data is collected over hourly interval thus having  $15 \times 24 = 360$  total instances. In our simulation setup, we consider interval size of 10 minutes, therefore we have used linear interpolation to expand the collected and total data instances are  $15 \times 24 \times 6 = 2160$ . Outdoor temperature, CO<sub>2</sub>, and relative humidity data along with user specified minimum and maximum settings for each parameter is given in Figure 6. Besides these three parameters, solar radiation and wind speed data is also presented in Figure 6 (d) as these two

external parameters are very crucial and have significant impact on indoor parameters.

Table 3 presents brief summary of collected data and user desired minimum and maximum setting for temperature, humidity and CO<sub>2</sub> level.

**B. NOISE MODELS FOR EVALUATING ACCURACY OF PREDICTION**

1) AVERAGE BASED MODEL

To create dynamically changing conditions, we have assumed variable error in sensor readings based on external parameters and actuators operational level using uniform distribution. The amount of error is randomly generated but its proportional to the average of accumulated error components for each parameter. In other words, noisy sensor readings are generated for temperature ( $sen_T$ ), humidity ( $sen_H$ ) and CO<sub>2</sub> ( $sen_C$ ) level using following expressions.

$$sen_T = GH_T + err_T \cdot N(-1, 1) \cdot S_T \tag{6}$$

$$sen_H = GH_H + err_H \cdot N(-1, 1) \cdot S_H \tag{7}$$

$$sen_C = GH_C + err_C \cdot N(-1, 1) \cdot S_C \tag{8}$$

where  $GH_T$ ,  $GH_H$ , and  $GH_C$  denotes the greenhouse actual temperature, humidity and CO<sub>2</sub> level.  $S_T$ ,  $S_H$ , and  $S_C$  is



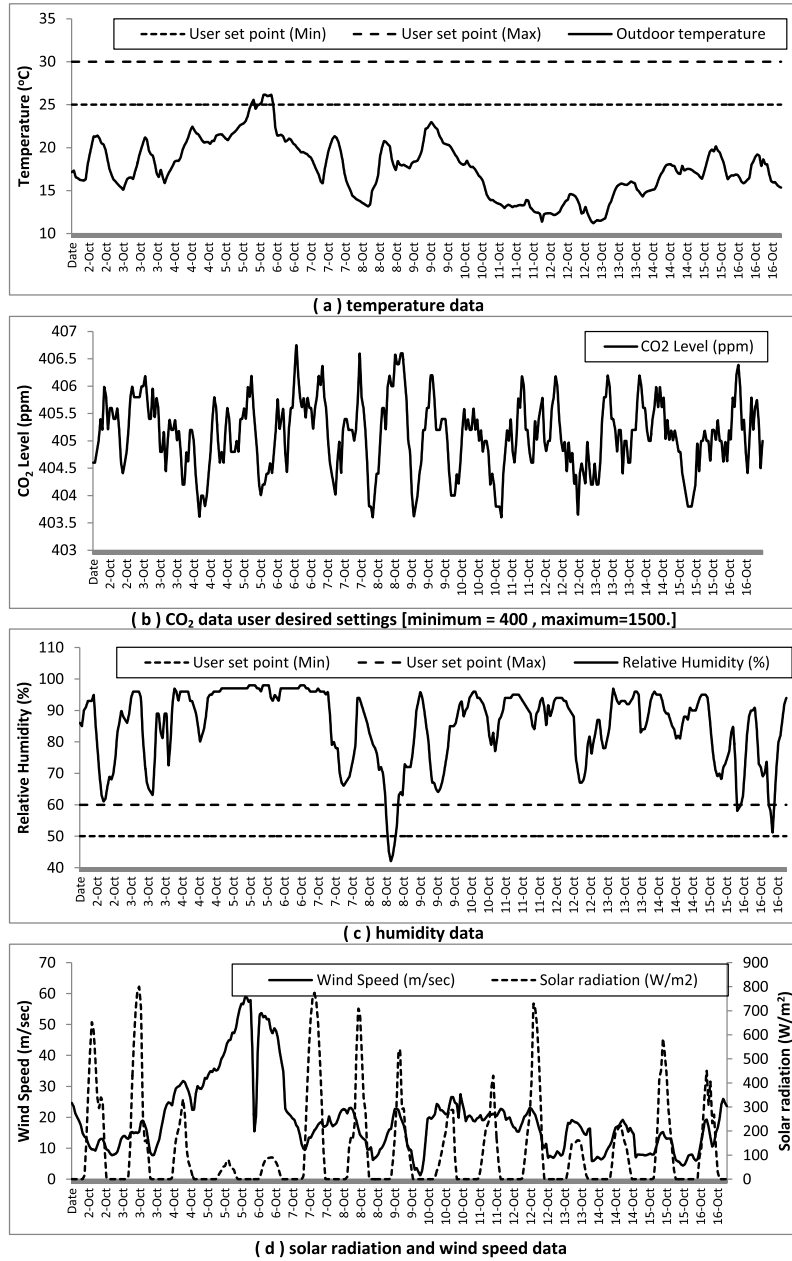


FIGURE 6. Outdoor environment data (15 days) with user desired settings.

TABLE 3. Summary of collected data used in simulation along with user desired settings.

Particulars		Temperature (°C)	Rel. Humidity (%)	CO <sub>2</sub> (ppm)	Solar Radiation (W/m <sup>2</sup> )	Wind Speed (m/sec)
Outdoor parameters values	Min	11.2	42.1	403.6	0	1.22
	Max	26.17	98	406.75	800.14	58.87
	Avg.	17.59	85.16	405.1	115.49	19.25
User desired settings	Min	25	50	400	NA	
	Max	30	60	1500		

error scaling factor for temperature, humidity and CO<sub>2</sub> level. Results reported in this study are collected with  $S_T = 10$ ,  $S_H = 10$ , and  $S_C = 100$ .  $N(-1, +1)$  is

a random number generator between  $-1$  and  $+1$  using uniform distribution. Accumulated error factor for temperature ( $err_T$ ), humidity ( $err_H$ ) and CO<sub>2</sub> level ( $err_C$ ) is

given by

$$err_T = \frac{eT + eS + eW + eNVLevel + eNVDur + eFVLevel}{11} + \frac{eFVDur + eHLevel + eHDur + eCLevel + eCDur}{11} \quad (9)$$

$$err_H = \frac{eH + eS + eW + eNVLevel + eNVDur + eFVLevel}{11} + \frac{eFVDur + eHumLevel + eHumDur + eFogLevel}{11} + \frac{eFogDur}{11} \quad (10)$$

$$err_C = \frac{eC + eS + eW + eNVLevel + eNVDur + eFVLevel}{9} + \frac{eFVDur + eCGenLevel + eCGenDur}{9} \quad (11)$$

where  $eT$ ,  $eH$ ,  $eC$ ,  $eS$ ,  $eW$ ,  $eNVLevel$ ,  $eNVDur$ ,  $eFVLevel$ ,  $eFVDur$ ,  $eHLevel$ ,  $eHDur$ ,  $eCLevel$ ,  $eCDur$ ,  $eHumLevel$ ,  $eHumDur$ ,  $eFogLevel$ ,  $eFogDur$ ,  $eCGenLevel$ ,  $eCGenDur$  are the normalized error factors due to indoor and outdoor temperature difference, indoor and outdoor humidity difference, indoor and outdoor  $CO_2$  level difference, solar radiation, wind speed, natural ventilation level and duration, forced ventilation level and duration, heater level and duration, chiller level and duration, dehumidifier level and duration, fogging system level and duration,  $CO_2$  generator level and duration, respectively. Table 4 presents the summary of resultant error measure due to above formulation in temperature,  $CO_2$  and humidity sensor readings collected from experimental results.

**TABLE 4. Error measures in temperature,  $CO_2$  and humidity sensor readings using average based model.**

Measure	Temperature	$CO_2$	Humidity
Min Error	-1.66	-34.74	-2.16
Max Error	2.11	26.13	2.5
Average Error	0.07	-0.07	0.03

## 2) MAXIMUM BASED MODEL

Initially, the average based model was used to generate noise in sensor readings. However, the resultant error was very small as shown in Table 4. Therefore, another set of experiments were conducted with maximum based error model to highlight the gain in accuracy by learning to prediction scheme. This is to create dynamically changing conditions, we have assumed variable error in sensor readings based on external parameters and actuators operational level using uniform distribution. Amount of error is randomly generated but its proportional to the maximum error components for each parameter. In other words, noisy sensor readings are generated for temperature ( $sen_T$ ), humidity ( $sen_H$ ) and  $CO_2$  ( $sen_C$ ) level using following expressions.

$$sen_T = GH_T + err_T \cdot N(-1, 1) \cdot S_T \quad (12)$$

$$sen_H = GH_H + err_H \cdot N(-1, 1) \cdot S_H \quad (13)$$

$$sen_C = GH_C + err_C \cdot N(-1, 1) \cdot S_C \quad (14)$$

where  $GH_T$ ,  $GH_H$ , and  $GH_C$  denotes the greenhouse actual temperature, humidity and  $CO_2$  level.  $S_T$ ,  $S_H$ , and  $S_C$  is error scaling factor for temperature, humidity and  $CO_2$  level. Results reported in this study with maximum based model are collected with  $S_T = 5$ ,  $S_H = 5$ , and  $S_C = 50$ .  $N(-1, +1)$  is a random number generator between  $-1$  and  $+1$  using uniform distribution. Accumulated error factor for temperature ( $err_T$ ), humidity ( $err_H$ ) and  $CO_2$  level ( $err_C$ ) is given by

$$err_T = \text{Max}(eT + eS + eW + eNVLevel + eNVDur + eFVLevel + eFVDur + eHLevel + eHDur + eCLevel + eCDur) \quad (15)$$

$$err_H = \text{Max}(eH + eS + eW + eNVLevel + eNVDur + eFVLevel + eFVDur + eHumLevel + eHumDur + eFogLevel + eFogDur) \quad (16)$$

$$err_C = \text{Max}(eC + eS + eW + eNVLevel + eNVDur + eFVLevel + eFVDur + eCGenLevel + eCGenDur) \quad (17)$$

Table 5 presents the summary of resultant error measure due to above formulation in temperature,  $CO_2$  and humidity sensor readings collected from experimental results.

**TABLE 5. Error measures in temperature,  $CO_2$  and humidity sensor readings using maximum based model.**

Measure	Temperature	$CO_2$	Humidity
Min Error	-5	-49	-5
Max Error	4	33	4
Average Error	0.05	-0.39	-0.26

## C. SIMULATION SCENARIOS FOR PERFORMANCE EVALUATION

Experimental results are collected for two different scenarios. In Case Study-A, error in sensor readings is generated using the average based model. In Case Study-B, error in sensor readings is generated using the maximum based model. Table 6 presents the summary configuration of various modules and experimental setup for the two selected scenarios.

## D. TRAINING AND TESTING OF LEARNING MODULES

For training and testing of learning modules, we have conducted repeated number of experiments and collected the simulation data. During simulation execution, we maintain log/trace files that holds various required data e.g. current indoor parameters values, corresponding actuator operational status and outdoor environmental conditions. In the following subsections, we briefly discuss about the collected data and results of training and testing of learning modules, separately.

Sample view of collected data for the training of temperature based Kalman filter learning module is provided in Table 7. This is just to give an idea about data distribution. Similarly, we have training dataset collected for training of learning modules for error prediction and tuning of Kalman

TABLE 6. Simulation scenarios for performance analysis.

Scenario	Scheme	Algorithm Used	Noise Model
Case Study-A	Baseline	-	Average based Model
	Prediction	Kalman filter	
	Learning to Prediction	Kalman filter with ANN	
Case Study-B	Baseline	-	Maximum based Model
	Prediction	Kalman filter	
	Learning to Prediction	Kalman filter with ANN	

TABLE 7. Sample data for training of ANN based learning module for temperature.

S. No.	Inputs												Output errinTemp
	Ext. Temp	Solar Radiation	Wind Speed	Sensor Temp	Natural Vent.		Forced Vent.		Heater		chiller		
					Level	Dur	Level	Dur	Level	Dur	Level	Dur	
1	17.16	0	24.63	26	10	10	0	0	0	0	0	0	-2.58
2	17.19	0	24.44	17	0	0	0	0	10	10	0	0	-2.76
3	17.22	0	24.26	23	0	0	9	10	0	0	0	0	-2.34
4	17.24	0	24.07	33	0	0	10	10	0	0	0	0	-2.88
5	17.27	0	23.89	26	0	0	10	10	0	0	0	0	-2.56
6	17.3	0	23.7	31	7	10	0	0	0	0	0	0	-2.51
7	17.33	0	23.51	34	0	0	0	0	0	0	0	0	-1.1
8	17.2	0	23.14	22	9	10	0	0	0	0	0	0	-2.28
9	17.07	0	22.77	24	0	0	0	0	0	0	0	0	-0.65
10	16.94	0	22.39	35	8	10	0	0	0	0	0	0	-2.78
...	...	...	...	...	...	...	...	...	...	...	...	...	...
10000	16.54	0	21.27	22	0	0	6	10	0	0	0	0	-2.01

filter modules for noise removal in CO<sub>2</sub> and humidity sensors readings.

In these experiments, we have used 75% of available data for training and 25% is used for testing. The training data in these experiments have 10000 records. During the training process, different settings of ANN algorithms are tested with different number of neurons in the hidden layer, changing learning rates and activation function using 4-fold cross-validation technique to eliminate biasness in training.

The comparison of the predicted and actual errors in sensors readings is illustrated in Figure 7 for training data. The results indicate that our proposed prediction algorithm performs well. In order to present the comparison results more effectively for better understanding, the absolute error has been calculated as shown in Figure 7. The absolute error values indicate that the majority of the values lies in the range of [-.5, +.5], [-2, +2], [-0.1, +0.1] for temperature, CO<sub>2</sub>, and humidity, respectively. The root mean square error (RMSE) for temperature is 0.067, for CO<sub>2</sub> is 1.324, and for humidity is 0.8. These values indicate that the performance of the proposed method is quite impressive. These results measure the performance of the model as well as make us confident to utilize the training model to predict the error and tune the related Kalman filter modules.

## VI. RESULTS AND DISCUSSION

### A. CASE-STUDY A

In this part, we have used average based model for simulating noisy sensor reading in greenhouse indoor environment due to calibration and locality errors. Figure 8 presents prediction results for temperature sensor reading of the three schemes. It should be noted that actual temperature reading indicate

the real sensor readings (without error). Furthermore, actual temperature sensor readings in each case looks different and this is because of the different optimization procedure and resultant actuators operations. Baseline scheme does not use any prediction algorithm and it simply make decision based on the noises sensor reading. In prediction scheme, temperature prediction results by Kalman filter (without learning) attempts to remove the fluctuation and noise from sensor readings and resultant graph is smoother as compared to baseline scheme. In the proposed learning to prediction scheme, Kalman filter algorithm is continuously tuned by learning module and its prediction results are further improved.

Figure 9 presents prediction results for CO<sub>2</sub> sensor reading of the three schemes. It should be noted that the actual CO<sub>2</sub> reading indicates the real sensor readings (without error). Furthermore, actual CO<sub>2</sub> sensor readings in each case looks different and this is because of the different optimization procedures and resultant actuators operations. Baseline scheme does not use any prediction algorithm and it simply makes decision based on the noisy sensor reading. In the prediction scheme, CO<sub>2</sub> prediction results by Kalman filter (without learning) attempts to remove the fluctuation and noise from sensor readings and resultant graph is smoother as compared to the baseline scheme. In the proposed learning to prediction scheme, the Kalman filter algorithm is continuously tuned by learning module and its prediction results are further improved.

Figure 10 presents prediction results for humidity sensor reading of the three schemes. It should be noted that actual humidity reading indicate the real sensor readings (without error). Furthermore, actual humidity sensor readings

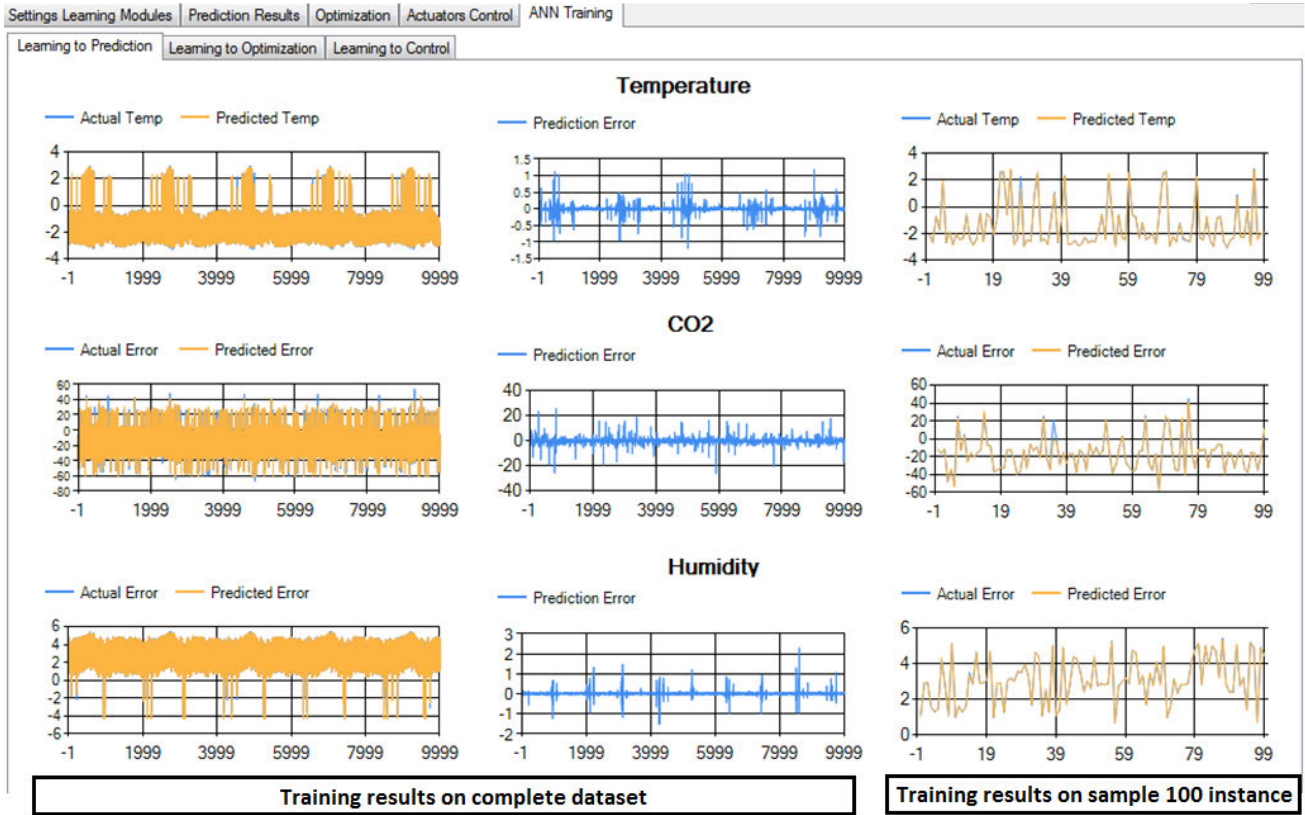


FIGURE 7. Testing results of learning modules for prediction part after training.

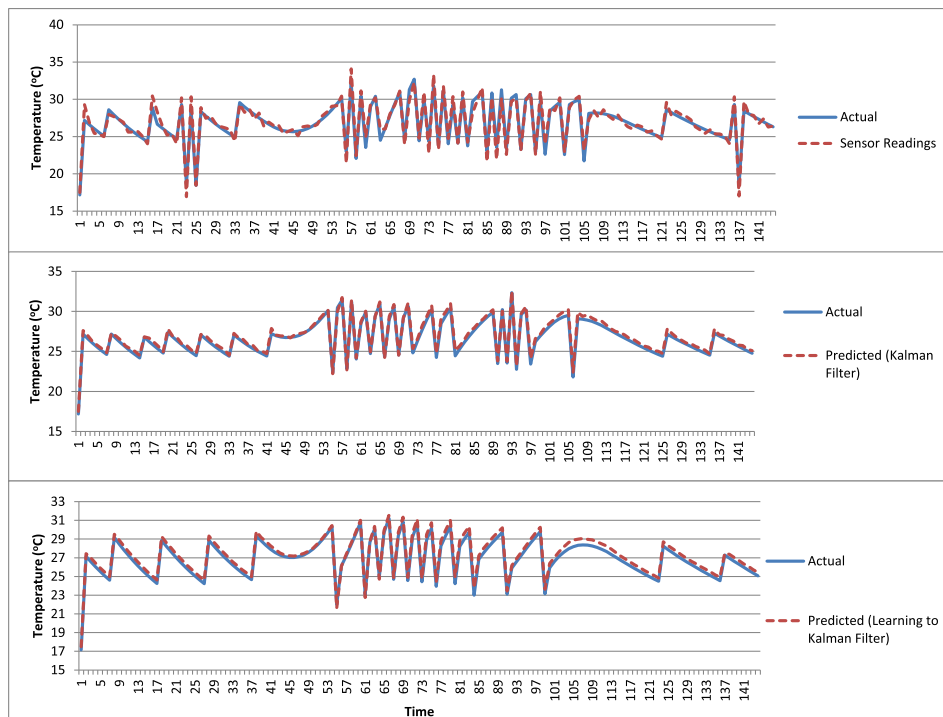


FIGURE 8. Temperature prediction results for the three schemes (Case Study-A).

in each case looks different and this is because of the different optimization procedure and resultant actuators operations. Baseline scheme does not use any prediction algorithm

and it simply makes decision based on the noises sensor reading. In prediction scheme, humidity prediction results by Kalman filter (without learning) attempts to remove the

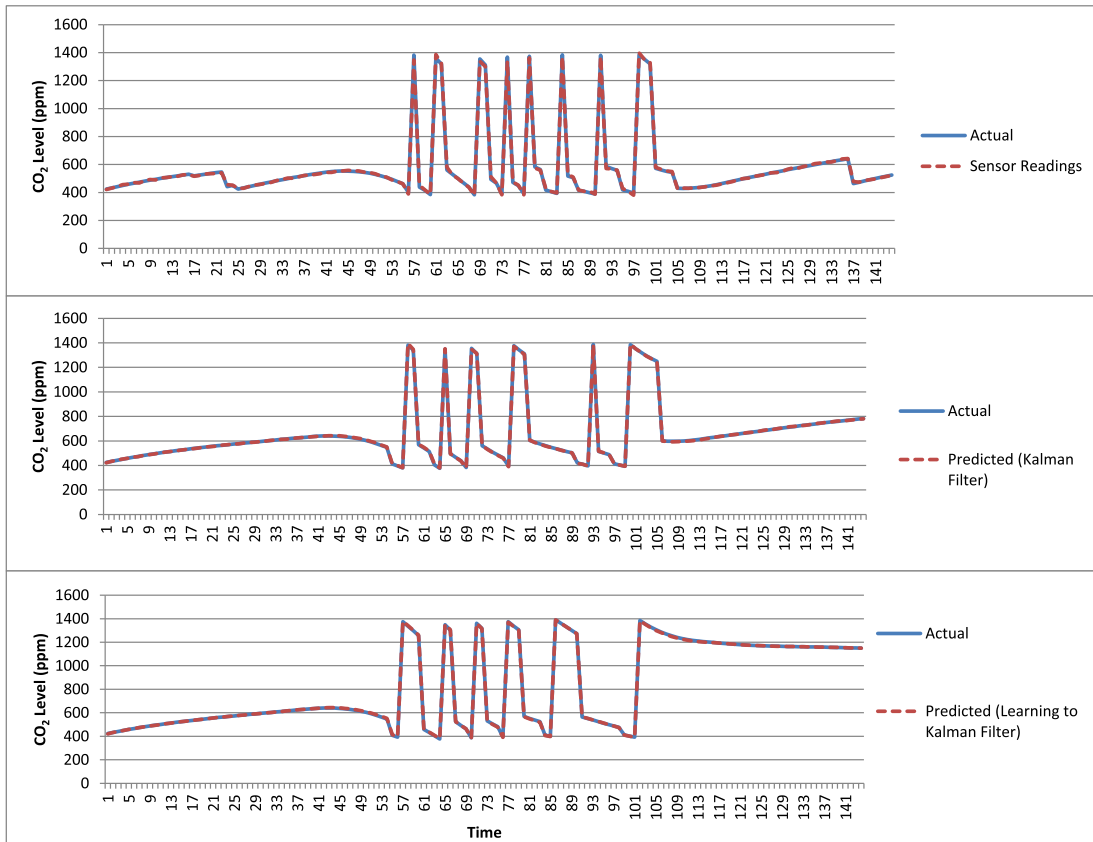


FIGURE 9. CO<sub>2</sub> prediction results for the three schemes (Case Study-A).

fluctuation and noise from sensor readings and resultant graph is smoother as compared to baseline scheme. In the proposed learning to prediction scheme, the Kalman filter algorithm is continuously tuned by learning module and its prediction results are further improved.

Comparative analysis of prediction part results for indoor parameters predictions from noisy sensor readings given in Figure 8, Figure 9, Figure 10 reveals that proposed learning to prediction scheme perform slightly better than the other two schemes in predicting the actual indoor parameters values from noisy sensor readings. However, the difference is not clearly visible in the graphical results, therefore we conduct statistical analysis of the prediction results using root mean squared error (RMSE) metric (Equation 18).

$$RMSE = \sqrt{\sum_{i=1}^n \frac{(P_i - \hat{P}_i)^2}{n}} \quad (18)$$

Statistical summary of the results for case-study A is presented in Table 8. Best results for Kalman filter algorithm (without learning module) were obtained with a fix value of  $R = 10$  and presented in this table. Similarly, best results for Kalman filter with ANN learning module are reported in this table with  $F = 0.01$ . Proposed learning to prediction model

TABLE 8. Statistical summary of prediction part results (Case Study-A).

Parameter	Scheme	RMSE
Temperature	Baseline Scheme	0.32
	Predicted (Kalman Filter)	0.18
	Predicted (Learning to Kalman Filter)	<b>0.15</b>
CO <sub>2</sub>	Baseline Scheme	3.21
	Predicted (Kalman Filter)	1.26
	Predicted (Learning to Kalman Filter)	<b>1.22</b>
Humidity	Baseline Scheme	0.35
	Predicted (Kalman Filter)	0.2
	Predicted (Learning to Kalman Filter)	<b>0.18</b>

results outperforms other two schemes for each of the selected greenhouse indoor parameter.

However, very low relative improvement can be observed in prediction accuracy of proposed learning to prediction model when compared to results of Kalman filter without learning module. This is due to the fact that average based model results in very low noise in sensor readings. We can see from Table 4 that absolute error in CO<sub>2</sub> is higher and therefore relative improvement of proposed scheme is also significant for CO<sub>2</sub> sensor readings.

### B. CASE-STUDY B

In this part, we have used maximum based model for generating noise in sensor readings. Figure 11 presents



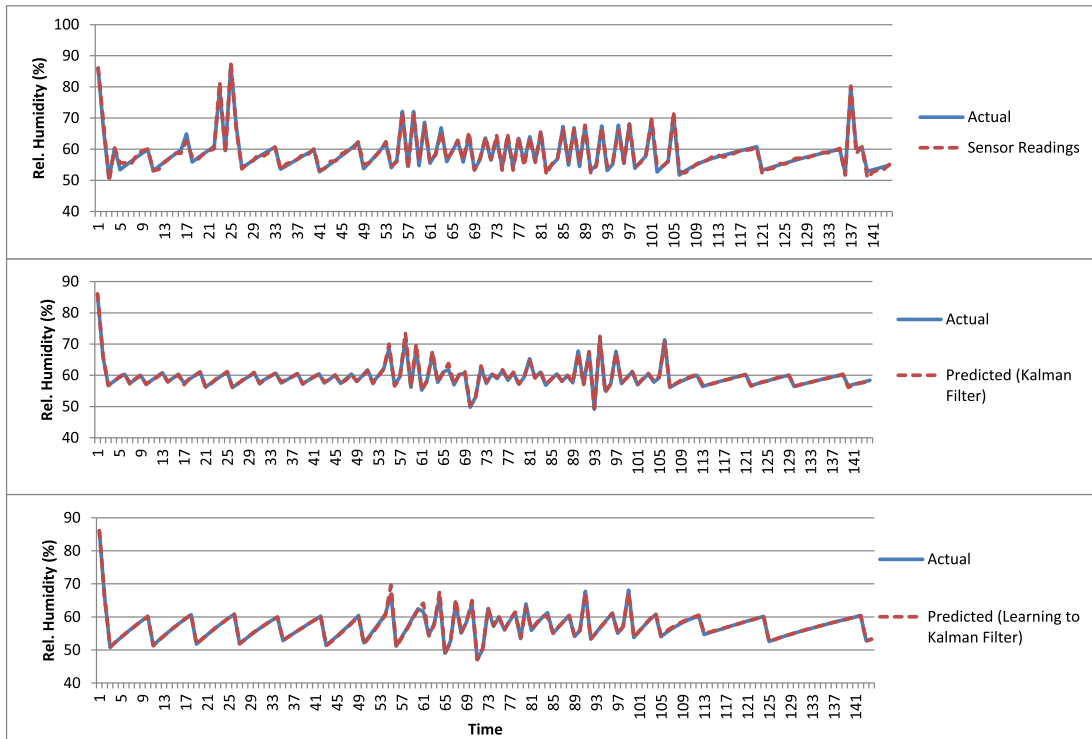


FIGURE 10. CO<sub>2</sub> prediction results for the three schemes (Case Study-A).

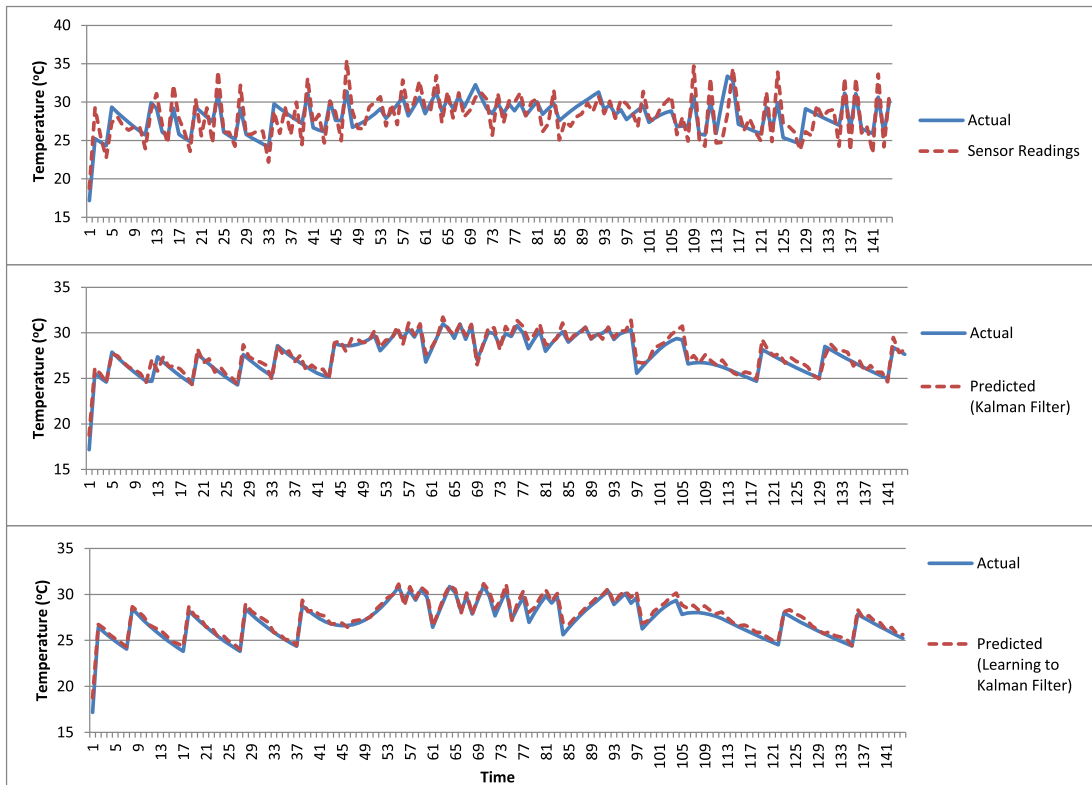


FIGURE 11. Temperature prediction results for the three schemes (Case Study-B).

prediction results for temperature sensor reading of the three schemes. It should be noted that the actual temperature reading indicates the real sensor readings (without error).

Furthermore, actual temperature sensor readings in each case looks different and this is because of the different optimization procedure and resultant actuators operations.

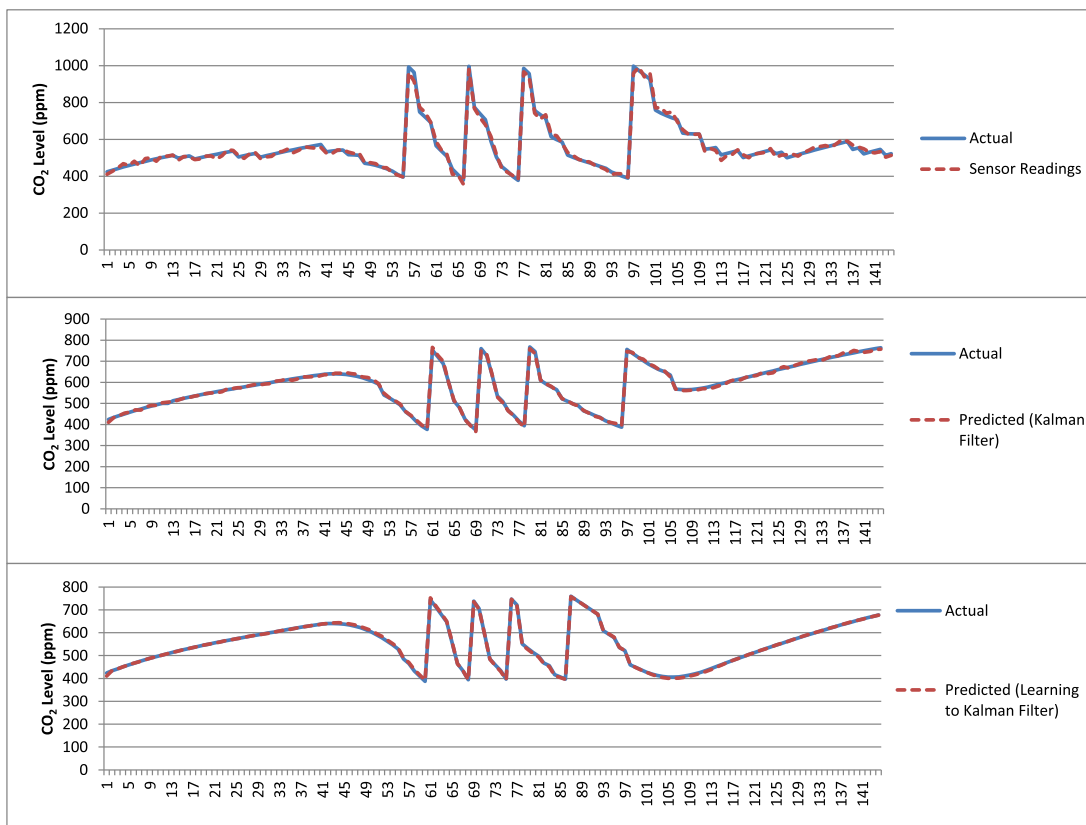


FIGURE 12. CO<sub>2</sub> prediction results for the three schemes (Case Study-B).

The baseline scheme does not use any prediction algorithm and it simply makes decision based on the noises sensor reading. These results have higher noise as compared to the results of Case Study-A with average based noise model. In the prediction scheme, temperature prediction results by the Kalman filter (without learning) attempts to remove the fluctuation and noise from sensor readings and resultant graph is smoother as compared to baseline scheme. In the proposed learning to optimization scheme, the Kalman filter algorithm is continuously tuned by learning module and its prediction results are further improved.

Figure 12 presents prediction results for CO<sub>2</sub> sensor reading of the three schemes. It should be noted that actual CO<sub>2</sub> reading indicate the real sensor readings (without error). Furthermore, actual CO<sub>2</sub> sensor readings in each case looks different and this is because of the different optimization procedure and resultant actuators operations. Baseline scheme does not use any prediction algorithm and it simply make decision based on the noises sensor reading. These results have higher noise as compared to Case Study-A results with average based noise model. In prediction scheme, temperature prediction results by the Kalman filter (without learning) attempts to remove the fluctuation and noise from sensor readings and resultant graph is smoother as compared to baseline scheme. In the proposed learning to prediction scheme,

the Kalman filter algorithm is continuously tuned by learning module and its prediction results are further improved.

Figure 13 presents prediction results for humidity sensor reading of the three schemes. It should be noted that actual humidity reading indicate the real sensor readings (without error). Furthermore, actual humidity sensor readings in each case looks different and this is because of the different optimization procedure and resultant actuators operations. Baseline scheme does not use any prediction algorithm and it simply makes decision based on the noises sensor reading. These results have higher noise as compared to Case Study-A results with average based noise model. In prediction scheme, temperature prediction results by the Kalman filter (without learning) attempts to remove the fluctuation and noise from sensor readings and resultant graph is smoother as compared to baseline scheme. In the proposed learning to prediction scheme, the Kalman filter algorithm is continuously tuned by learning module and its prediction results are further improved.

Comparative analysis of prediction part results for indoor parameters predictions from noisy sensor readings given in Figure 11, Figure 12, and Figure 13 reveals that proposed learning to prediction scheme perform slightly better than the other two schemes in predicting the actual indoor parameters values from noisy sensor readings. Due to relatively higher

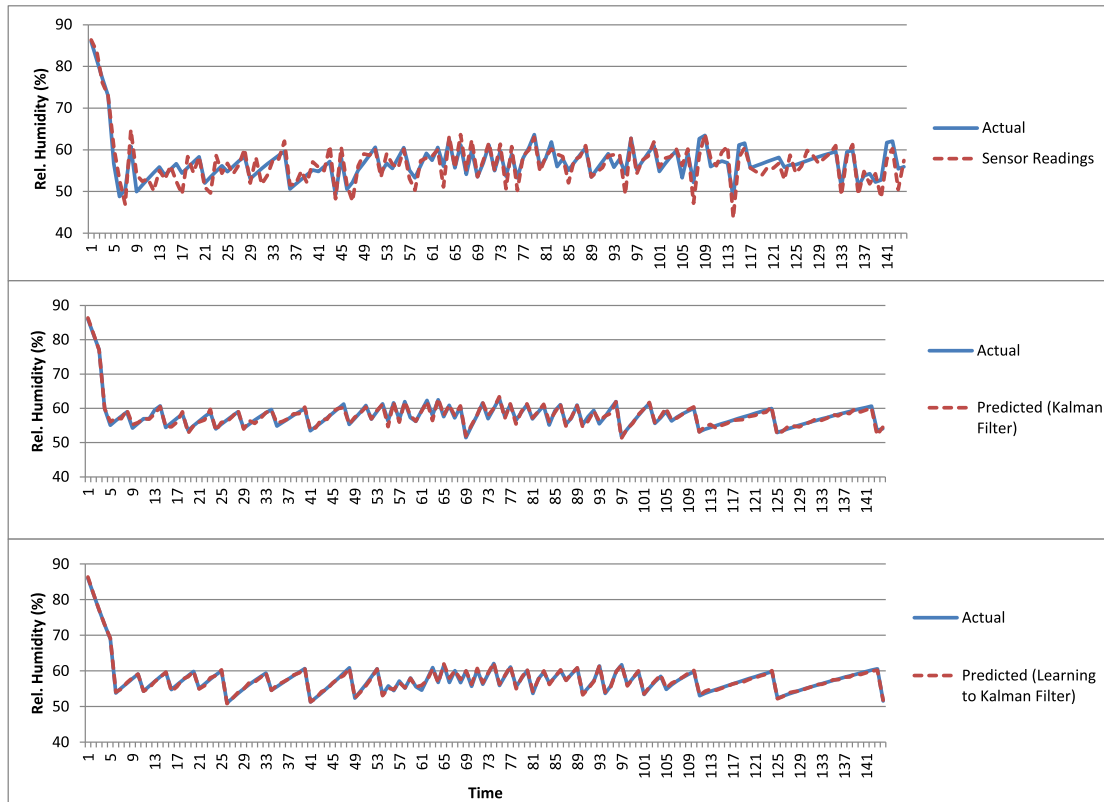


FIGURE 13. Humidity prediction results for the three schemes (Case Study-B).

TABLE 9. Statistical summary of prediction part results (Case Study-B).

Parameter	Scheme	RMSE
Temperature	Baseline Scheme	0.8
	Predicted (Kalman Filter)	0.26
	Predicted (Learning to Kalman Filter)	<b>0.2</b>
CO <sub>2</sub>	Baseline Scheme	6.26
	Predicted (Kalman Filter)	2.27
	Predicted (Learning to Kalman Filter)	<b>1.29</b>
Humidity	Baseline Scheme	0.98
	Predicted (Kalman Filter)	0.25
	Predicted (Learning to Kalman Filter)	<b>0.14</b>

error rate, the difference is more visible in the graphical results as compared to Case Study-A results. For further quantification, we conduct statistical analysis of the prediction results using root mean squared error metric given in Equation 18. Statistical measures are used for quantifiable comparative analysis by summarizing the results in the form of a single statistical value. Statistical summary of the results for case-study B is presented in Table 9. Best results for Kalman filter algorithm (without learning module) were obtained with a fix value of  $R = 10$  and presented in this table. Similarly, best results for Kalman filter with ANN learning module are reported in this table with  $F = 0.1$ . Proposed learning to prediction model again outperforms other two schemes for each of the selected greenhouse indoor parameter.

Contrary to the case-study A, significant relative improvement can be observed in prediction accuracy of proposed

learning to prediction model when compared to results of Kalman filter without learning module. This is due to the fact that maximum based model results in relatively high noise in sensor readings. We can see from Table 5 that absolute error in all parameters is much higher as compared to average based model and therefore relative improvement of proposed scheme is also significant for all sensor readings.

### VII. CONCLUSION AND FUTURE WORK

A novel model based on learning to prediction is presented in this article to improve the accuracy of prediction algorithms in dynamically changing conditions. Conventional prediction algorithms are locked after training and they fail to adapt with changing operational conditions. To address this limitation, the proposed learning to prediction model continuously monitor the environment and performance of the prediction algorithm and when triggers are observed then internal parameters of prediction algorithm are tuned. For experimental analysis of the proposed model, we have considered a greenhouse indoor environment where the Kalman filter algorithm prediction accuracy is degraded due to changing external environmental weather and internal operational condition of various actuators. A learning module based on ANN is used to improve the performance of the Kalman filter algorithm by tuning its parameter  $R$ . In this study, separate instance of Kalman filter algorithm is used for each selected indoor parameter i.e. temperature, CO<sub>2</sub> and humidity. With each Kalman

filter instance, we have a learning module based on ANN. These learning modules take appropriate parameters as input such as external weather and internal actuator operational status, to predict estimated error  $err$  in sensor readings as output. Afterwards,  $err$  is divided by a constant factor  $F$  to obtain updated value of  $R$  in corresponding Kalman filter module which is further used in computation of Kalman gain  $K$ . Experiments are conducted with different noise models to generate noise in sensor reading i.e. average and maximum based noise models. Results of proposed learning model are compared with baseline scheme and conventional Kalman filter algorithm. Comparative analysis of the results clearly indicate that proposed learning to prediction model have significantly improved the accuracy of Kalman filter algorithm. Furthermore, it was also observed that performance gain in prediction accuracy is directly proportional to the amount of noise introduced in sensor readings. By comparing the accuracy of proposed model based on learning with conventional Kalman filter algorithm in terms of RMSE metric, relative improvement in results is 16.66% and 23.07% for temperature, 3.17% and 47.17% for  $CO_2$ , and 10% and 44% for humidity, with average and maximum based error models, respectively. In the future, we are looking forward to extend this study in two directions (a) use deep learning algorithms (instead of ANN) to tune the performance of other prediction algorithms (b) conduct experimental analysis with big data in more complex real world applications to further establish the validity of proposed learning to prediction mechanism.

## ACKNOWLEDGMENT

Any correspondence related to this article should be addressed to Dohyeun Kim.

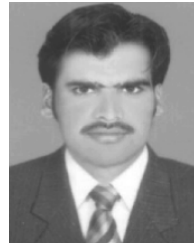
## REFERENCES

- [1] *World Population Prospects 2017*, United Nations, New York, NY, USA, 2017.
- [2] *The Future of Food and Agriculture—Trends and Challenges*, FAO, Rome, Italy, 2017.
- [3] I. Fao. (2017). Wfp, who (2017) the state of food security and nutrition in the world 2017, Building resilience for peace and food security. FAO, Rome. Accessed: May 16, 2018. [Online]. Available: <http://www.fao.org/3/a-i7695e.pdf>
- [4] *Sustainable Development Goals, SDGs, Transforming our World: The, Volume 2030*, G. Assembly, New York, NY, USA, 2015.
- [5] J.-C. Pomerol, "Artificial intelligence and human decision making," *Eur. J. Oper. Res.*, vol. 99, no. 1, pp. 3–25, 1997.
- [6] I. Ullah, M. Fayaz, and D. Kim, "Improving accuracy of the Kalman filter algorithm in dynamic conditions using ANN-based learning module," *Symmetry*, vol. 11, no. 1, p. 94, Jan. 2019.
- [7] M. A. Akkaş and R. Sokullu, "An IoT-based greenhouse monitoring system with Micaz nodes," *Procedia Comput. Sci.*, vol. 113, pp. 603–608, Jan. 2017.
- [8] Y. Shen, R. Wei, and L. Xu, "Energy consumption prediction of a greenhouse and optimization of daily average temperature," *Energies*, vol. 11, no. 1, p. 65, Jan. 2018.
- [9] J. Chen, J. Zhao, F. Xu, H. Hu, Q. Ai, and J. Yang, "Modeling of energy demand in the greenhouse using PSO-GA hybrid algorithms," *Math. Problems Eng.*, vol. 2015, pp. 1–6, Feb. 2015.
- [10] M. Trejo-Perea, G. Herrera-Ruiz, J. Rios-Moreno, R. C. Miranda, and E. Rivasaraiza, "Greenhouse energy consumption prediction using neural networks models," *Training*, vol. 1, no. 1, p. 2, 2009.
- [11] D. Yates, S. Gangopadhyay, B. Rajagopalan, and K. Strzepak, "A technique for generating regional climate scenarios using a nearest-neighbor algorithm," *Water Resour. Res.*, vol. 39, no. 7, pp. 1–15, Jul. 2003.
- [12] M.-L. Zhang and Z.-H. Zhou, "A k-nearest neighbor based algorithm for multi-label classification," in *Proc. IEEE Int. Conf. Granular Comput.*, vol. 2, Dec. 2005, pp. 718–721.
- [13] L. Breiman, *Classification and Regression Trees*. Evanston, IL, USA: Routledge, 2017.
- [14] M. Slocum, "Decision making using id3 algorithm," *Insight, River Academic J.*, vol. 8, no. 2, pp. 1–12, 2012.
- [15] J. R. Quinlan, *C4. 5: Programs for Machine Learning*. Amsterdam, The Netherlands: Elsevier, 2014.
- [16] M. van Diepen and P. H. Franses, "Evaluating chi-squared automatic interaction detection," *Inf. Syst.*, vol. 31, no. 8, pp. 814–831, Dec. 2006.
- [17] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.
- [18] G. Zhang, B. E. Patuwo, and M. Y. Hu, "Forecasting with artificial neural networks: The state of the art," *Int. J. Forecasting*, vol. 14, pp. 35–62, Mar. 1998.
- [19] G. Merkel, R. Povinelli, and R. Brown, "Short-term load forecasting of natural gas with deep neural network regression," *Energies*, vol. 11, no. 8, p. 2008, Aug. 2018.
- [20] N. Baykan and N. Yilmaz, "A mineral classification system with multiple artificial neural network using K-fold cross validation," *Math. Comput. Appl.*, vol. 16, no. 1, pp. 22–30, Apr. 2011.
- [21] K. Genikomsakis, S. Lopez, P. Dallas, and C. Ioakimidis, "Simulation of wind-battery microgrid based on short-term wind power forecasting," *Appl. Sci.*, vol. 7, no. 11, p. 1142, Nov. 2017.
- [22] D. Afolabi, S.-U. Guan, K. Man, P. Wong, and X. Zhao, "Hierarchical meta-learning in time series forecasting for improved interference-less machine learning," *Symmetry*, vol. 9, no. 11, p. 283, Nov. 2017.
- [23] S. Sathyanarayana, "A gentle introduction to backpropagation," *Numeric Insight*, vol. 7, pp. 1–15, Jul. 2014.
- [24] S. Lai, L. Xu, K. Liu, and J. Zhao, "Recurrent convolutional neural networks for text classification," in *Proc. AAAI*, vol. 333, 2015, pp. 2267–2273.
- [25] X. Zhang and Y. LeCun, "Text understanding from scratch," 2015, *arXiv:1502.01710*. [Online]. Available: <http://arxiv.org/abs/1502.01710>
- [26] Y. Kim, "Convolutional neural networks for sentence classification," 2014, *arXiv:1408.5882*. [Online]. Available: <http://arxiv.org/abs/1408.5882>
- [27] H. Sak, A. Senior, and F. Beaufays, "Long short-term memory recurrent neural network architectures for large scale acoustic modeling," in *Proc. 115th Annu. Conf. Int. Speech Commun. Assoc.*, 2014, pp. 1–5.
- [28] Z.-H. Zhou, J. Wu, and W. Tang, "Ensembling neural networks: Many could be better than all," *Artif. Intell.*, vol. 137, nos. 1–2, pp. 239–263, May 2002.
- [29] A. I. Naimi and L. B. Balzer, "Stacked generalization: An introduction to super learning," *Eur. J. Epidemiol.*, vol. 33, no. 5, pp. 459–464, May 2018.
- [30] D. H. Wolpert, "Stacked generalization," *Neural Netw.*, vol. 5, no. 2, pp. 241–259, Jan. 1992.
- [31] R. A. Jacobs, "Methods for combining experts' probability assessments," *Neural Comput.*, vol. 7, no. 5, pp. 867–888, Sep. 1995.
- [32] W. Qiao, Z. Yang, Z. Kang, and Z. Pan, "Short-term natural gas consumption prediction based on volterra adaptive filter and improved whale optimization algorithm," *Eng. Appl. Artif. Intell.*, vol. 87, Jan. 2020, Art. no. 103323.
- [33] W. Qiao, H. Moayedi, and L. K. Foong, "Nature-inspired hybrid techniques of IWO, DA, ES, GA, and ICA, validated through a k-fold validation process predicting monthly natural gas consumption," *Energy Buildings*, vol. 217, Jun. 2020, Art. no. 110023.
- [34] W. Qiao, K. Huang, M. Azimi, and S. Han, "A novel hybrid prediction model for hourly gas consumption in supply side based on improved whale optimization algorithm and relevance vector machine," *IEEE Access*, vol. 7, pp. 88218–88230, 2019.
- [35] W. Qiao and Z. Yang, "Forecast the electricity price of U.S. using a wavelet transform-based hybrid model," *Energy*, vol. 193, Feb. 2020, Art. no. 116704.
- [36] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, G. van den Driessche, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel, and D. Hassabis, "Mastering the game of Go with deep neural networks and tree search," *Nature*, vol. 529, no. 7587, p. 484, 2016.

- [37] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, M. G. Bellemare, I. Antonoglou, H. King, D. Kumaran, S. Legg, D. Wierstra, and D. Hassabis, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, p. 529, 2015.
- [38] C. Liu, G. Wen, Z. Zhao, and R. Sedaghati, "Neural-network-based sliding-mode control of an uncertain robot using dynamic model approximated switching gain," *IEEE Trans. Cybern.*, early access, Mar. 18, 2020, doi: [10.1109/TCYB.2020.2978003](https://doi.org/10.1109/TCYB.2020.2978003).
- [39] Z. Zhao, J. Shi, X. Lan, X. Wang, and J. Yang, "Adaptive neural network control of a flexible string system with non-symmetric dead-zone and output constraint," *Neurocomputing*, vol. 283, pp. 1–8, Mar. 2018.
- [40] Z. Zhao, X. Wang, C. Zhang, Z. Liu, and J. Yang, "Neural network based boundary control of a vibrating string system with input deadzone," *Neurocomputing*, vol. 275, pp. 1021–1027, Jan. 2018.
- [41] C. W. Kang and C. G. Park, "Attitude estimation with accelerometers and gyros using fuzzy tuned Kalman filter," in *Proc. Eur. Control Conf. (ECC)*, Aug. 2009, pp. 3713–3718.
- [42] M. N. Ibarra-Bonilla, P. J. Escamilla-Ambrosio, and J. M. Ramirez-Cortes, "Attitude estimation using a neuro-fuzzy tuning based adaptive Kalman filter," *J. Intell. Fuzzy Syst.*, vol. 29, no. 2, pp. 479–488, Oct. 2015.
- [43] H. Rong, C. Peng, Y. Chen, L. Zou, Y. Zhu, and J. Lv, "Adaptive-gain regulation of extended Kalman filter for use in inertial and magnetic units based on hidden Markov model," *IEEE Sensors J.*, vol. 18, no. 7, pp. 3016–3027, Apr. 2018.
- [44] R. E. Kalman, "A new approach to linear filtering and prediction problems," *J. Basic Eng.*, vol. 82, no. 1, pp. 35–45, Mar. 1960.
- [45] S. J. Julier and J. K. Uhlmann, "New extension of the Kalman filter to nonlinear systems," *Proc. SPIE*, vol. 3068, pp. 182–193, Apr. 1997.
- [46] C. R. Souza. (2014). *The Accord.NET Framework*. Accessed: Aug. 20, 2018. [Online]. Available: <http://accord-framework.net>
- [47] Meteoblue. (2018). *Weather Jeju City*. Accessed: Oct. 1, 2018. [Online]. Available: [https://www.meteoblue.com/en/weather/forecast/week/jeju-city\\_republic-of-korea\\_1846266](https://www.meteoblue.com/en/weather/forecast/week/jeju-city_republic-of-korea_1846266)



**ISRAR ULLAH** received the M.C.S. degree from the Institute of Computing and Information Technology (ICIT), Gomal University, Pakistan, in 2004, the M.S. degree in computer science from the National University of Computer and Emerging Sciences (NUCES), Islamabad, Pakistan, in 2009, and the Ph.D. degree in computer engineering from Jeju National University, South Korea, in 2019. His research work is focused on the application of prediction and optimization algorithms to the build IoT-based solutions. His research interests mainly include analytical modeling, network simulation, and analysis of optimization algorithms.



**MUHAMMAD FAYAZ** received the M.S. degree in computer science from SZABIST, Islamabad, Pakistan, in 2014, the M.Sc. degree in computer science from the University of Malakand, Chakdara, Pakistan, and the Ph.D. degree from the Department of Computer Engineering, Jeju National University, South Korea, in 2019.



**NASIR NAVEED** graduated from the Institute of Web Science and Technologies, University of Koblenz and Landau, Germany, in 2013. He worked on temporal analysis of social media contents in the popular social networks using machine learning and information retrieval techniques. He is currently investigating the use of machine learning techniques for exploiting the text contents for search in linked open data and developing scalable methods for big data analysis and data science.



**DOHYEUN KIM** received the B.S. degree in electronics engineering and the M.S. and Ph.D. degrees in information telecommunication from Kyungpook National University, South Korea, in 1988, 1990, and 2000, respectively. He was with the Agency of Defense Development (ADD), from 1990 to 1995. Since 2004, he has been with Jeju National University, South Korea, where he is currently a Professor with the Department of Computer Engineering. From 2008 to 2009, he was a Visiting Researcher with the Queensland University of Technology, Australia. His research interests include sensor networks, M2M/IoT, energy optimization and prediction, intelligent service, and mobile computing.

...