IAES International Journal of Artificial Intelligence (IJ-AI)
Vol. 13, No. 2, June 2024, pp. 1658~1667
ISSN: 2252-8938, DOI: 10.11591/ijai.v13.i2.pp1658-1667

Design of smoke detection system using deep learning and sensor fusion with recursive feature elimination cross-validation

James Julian[1], Annastya Bagas Dewantara[2], Fitri Wahyuni[1]
[1]Department of Mechanical Engineering, Faculty of Engineering, University of Pembangunan Nasional Veteran, Jakarta, Indonesia
[2]Department of Electrical Engineering, Faculty of Engineering, University of Pembangunan Nasional Veteran, Jakarta, Indonesia

Article Info

Article history:

Received Apr 3, 2023
Revised Aug 30, 2023
Accepted Sep 30, 2023

Keywords:

Cross-validation
Deep learning
Feature selection
Sensor fusion
Smoke sensor

ABSTRACT

The fire safety system is an important component that controls material and immaterial losses. Fire disasters are generally indicated by the appearance of excess smoke and changes in temperature, pressure, and changes in other parameters in the environment. Conventional smoke sensors are limited in reading parameter changes around their environment, making them less effective in early fire detection. This study aims to design a smoke detection system as an early fire detection system, using sensor fusion based on deep learning using the recursive feature elimination method with cross-validation (RFECV) using a random forest classifier used to select optimal parameters from public datasets as the basis for determining the sensor to be used. Based on the RFECV optimal feature, a deep learning algorithm was performed and obtained an accuracy of 0.99, a precision of 0.99, a recall of 1.00, and an F1 score of 0.99, with a latency time of 34.02 µs, which is 71.76% times faster than the original model.

Corresponding Author:

James Julian
Department of Mechanical Engineering, Faculty of Engineering
University of Pembangunan Nasional Veteran Jakarta
St. RS Fatmawati, Pondok Labu, Cilandak, Central Jakarta, Indonesian
Email: zames@upnvj.ac.id

1. INTRODUCTION

The fire safety system is an essential component that plays a role in controlling the level of material sand immaterial losses. The World Health Organization (WHO) estimates at least 300,000 deaths due to accidents caused by burns, with 95% of these accidents originating from low- and middle-income countries [1]. Fires are dominated by residential and office areas, with the most significant cause of non-fatal fires caused by cooking activities, namely 45%, and the most significant cause of fatal fires caused by smoking activities, namely 16% [2].

Fire disasters are generally indicated by the appearance of excess smoke and changes in temperature in the environment. Smoke sensors that are commonly used have limitations on the level of sensitivity so that there is a minimum volume of smoke to be detected as a fire; ionization-based smoke sensors have limits on the amount of smoke detected so that they are less effective in early detection of smoldering fires, and on photoelectric-based smoke, sensors have limitations on the detection activation time and the type of smoke that passes through it [3]. Futhermore, smoke sensors often give false positives when detecting a fire, such as when cooking steam is cooking and dust when cleaning [4]. Due to this limitation, it is necessary to have a more accurate and precise sensosr for detecting the presence of smoke as a fire prevention measure.

Several studies have been conducted to overcome this problem, one of which is using convolutional neural network (CNN)-based image processing on embedded systems [5], however, the detection system approach through pixel color readings often gives false positives when there is excessive lighting or contrast in the pixel readings. Hence, this approach is not suitable for use in indoor areas. Multiple wireless sensors are also used in a fire detection system using barometric sensors, humidity sensors, and temperature sensors connected to a server as a monitoring system [6]. This research shows that applying multiple sensors can be used as an indicator in determining forest fires. Still, this approach does not utilize the sensor fusion concept, which is capable of detecting fires. Still, these sensors act independently as parameter readings which require operator supervision in determining the occurrence of a fire hazard in the forest, The use of a machine learning approach with the supporting vector machine (SVM) method and partial least square discriminant analysis (PLS-DA) using gas-based sensors such as metal oxide sensors, electrochemical sensors, temperature sensors, and humidity, shows increased accuracy in the use of machine learning based on sensor fusion and gas parameter correlation in fire detection systems, the use of a barometer to measure air pressure around the fire area as a fire extinguisher, and the use of particulate-matter sensors to detect fires in wild forests [7]–[10]. The use of parameters such as heat, smoke and flame are also used as decision-making parameters in the sprinkler and alerting process [11], [12], and android based application with global positioning systems is also used for wildfire reporting and monitoring system [13]. The diversity of sensor types used as parameter reading instruments in early fire prevention systems has led to increased complexity in system design and the selection process of sensor components.

This study addresses the challenges by developing an advanced early fire prevention system. It employs a data-driven approach, utilizing recursive feature elimination with cross-validation (RFECV) based on sensor fusion and machine learning within a computer modeling program. Key parameters, including temperature, humidity, total volatile organic compounds (TVOC), eCO2, H2, Ethanol, and air pressure, are optimized through parameter selection. This optimization enables efficient smoke-related predictions. The RFECV outcomes are refined via deep neural network retraining on a microcontroller. This step serves as a test to assess the efficacy and efficiency of the system as an early fire prevention mechanism. Through this methodology, the paper aims to mitigate these issues and contribute to enhancing fire safety systems using a combination of data-driven techniques and advanced hardware for more effective fire prediction and prevention.

2. METHOD

The smoke detection system is an instrument used to detect smoke in the surrounding area. In conventional sensors, the presence of smoke is only detected based on the input value of one sensor parameter, so the tool's reliability is often questioned because smoke detection sensors often make false alarms or false positives where the alarm sounds even though there is no smoke around it. On this basis, it is necessary to add input values to increase the predictive ability of the smoke detection system.

This study discusses the design and conceptualization of the system using sensor fusion to combine inputs from several sensors to obtain input values from pressure, particular matter, humidity, temperature, TVOC, eCO2, and H2 which has been tested and can be used on microcontroller systems such as ESP8266 [14], [15]. The input values of these parameters will be used to predict the smoke's presence. In making the design, several steps need to be carried out to ensure that the input data can be used to effectively predict the presence of smoke. Shown in Figure 1. The process stages are divided into data preprocessing, feature selection, and modeling after the model is obtained, the model performance is evaluated to determine performance based on performance metrics and predictive latency time, the design of the new system can be carried out by paying attention to several aspects.

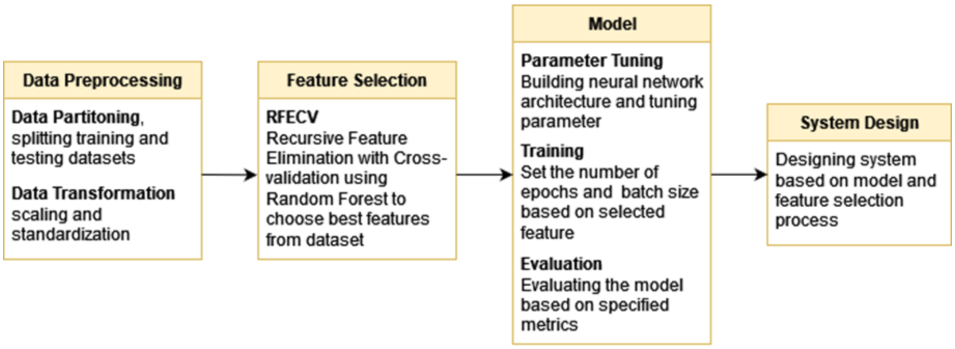


Figure 1. Data pipeline process

2.1. Dataset
The data used in this study uses secondary data from reading information from several sensors located on the Arduino Nicla Sense ME microcontroller architecture. BMP390, BME688, and SPG30 sensors are used to read humidity, temperature, air pressure, particulate matter, number of concentration particle and TVOC data. The data is sourced from public datasets [16] with the amount of data reaching 814,190 data with a sample rate of 1 Hz. Data was collected in indoor areas with high humidity, providing smoke from burning wood, coal, and gas grills.

2.2. RFECV
The features of the dataset have a lot of influence on computational time and computational size during the inference and training processes and the use of certain algorithms can speed up and improve the quality of the data used [17]. To speed up computational time and weight, it is necessary to eliminate redundant features that have little effect on the model. The method of eliminating this feature needs to pay attention to certain factors; this is intended so that the removed feature does not significantly influence the level of accuracy of the model created. The use of RFECV with the wrapper method has advantages over other methods; this is because the RFECV elimination process is based on a predictive model and does not depend on the dataset; the selection of this elimination is based on interactions between features so that the least significant features can be identified. The use of cross-validation is done to avoid the process of overfitting the model [18].

The RFECV procedure involves selecting a model as a reference for feature selection accuracy and subsequently selecting k-values to partition the dataset into k-segments. In this instance, the random forest classifier is adopted as the RFECV model. This choice is attributed to the classifier's adeptness in achieving high accuracy with constrained datasets, all while maintaining swift training times. Notably, this efficiency surpasses that of decision trees, support vector machines, and logistic regression classifiers [19].

2.3. Deep learning
The algorithm used in the prediction process is a deep learning algorithm; this is because deep learning algorithms have algorithms with the highest accuracy compared to other algorithms if the dataset is high [20]. Deep neural network is a machine learning subfield with characteristics like neural networks, often referred to as artificial neural networks. Deep neural network works by extracting features from a large amount of data by building a layered architecture between neuron nodes to form output neuron nodes that function as regression or classification as shown in Figure 2 [21].

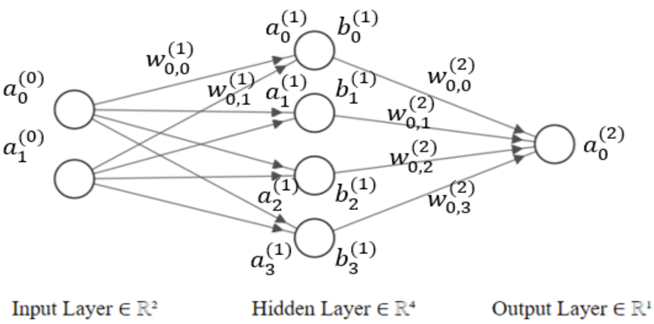


Figure 2. Deep neural network architecture

Forward propagation is a process of computing data from input data values in the initial neuron layer to producing output values in the final layer neurons which function to produce regression or classification results by adding the weight and bias values of each parameter in the deep neural network [22]. The value of the final output node is determined from the value of the neuron nodes in each layer. Each node is affected by the weight and bias values of the previous neuron node, which is then fed into the predetermined activation function [23], the equation of each neuron in each layer is represented in (1), with the value of (2) being the value of each parameter weight and bias that has been activated in (1).

$$z_j^L = \sum_{k=1}^{K} w_{j,k}^L a_k^{L-1} + b_j^L \quad (1)$$

$$a_j^{(L)} = \sigma\left(z_j^{(L)}\right) \quad (2)$$

L : Layer
k : Node input
K : Total node input
j : Node output
σ : Activation functions

The activation function is essential in the learning process in studying patterns and complex relationships from data to given labels. The activation function is tasked with obtaining the final value of the node obtained by entering the weight and bias values of the initial neuron node into the specified activation function. The rectified linear unit (ReLU) activation function is the most widely used activation function in the hidden layer because of its light computational ability and ability to overcome vanishing gradients [24]. The equation of the ReLU activation function is shown in (4), with the resulting range of activation values (3); this equation is used in the training process in the form of a derivation in (5).

$$\sigma(z_j^L) = \begin{cases} 0 & for\ z_j^L < 0 \\ z_j^L & for z_j^L \geq 0 \end{cases} \tag{3}$$

$$\sigma(z_j^L) = max(0, z_j^L) \tag{4}$$

$$\sigma'(z_j^L) = \frac{\delta\sigma(z_j^L)}{\delta z_j^L} = u(z_j^L) \tag{5}$$

In binary classification, the activation function is used to map the value of the input node into two output values [25]. These two output values are divided by applying a threshold of 0.5 at the output node shown in (6). The initial calculation and declaration of the derivation of the final activation function are carried out using the chain rule to speed up computation time in the optimization process, as shown in (7).

$$\sigma(z_j^L) = \frac{1}{1+e^{-z_j^L}} \tag{6}$$

$$\frac{\delta\sigma(z_j^L)}{\delta z_j^L} = \frac{\delta}{\delta z_j^L}\left(\frac{1}{1+e^{-z_j^L}}\right) = \left(\frac{1}{1+e^{-z_j^L}}\right)\left(1 - \frac{1}{1+e^{-z_j^L}}\right) \tag{7}$$

The training process of the deep neural network is called the deep learning process; this process calculates the error between the output values of the final layer neurons to the actual output values of the data; this learning process is called backward propagation [26]. Backward propagation is evaluated by changing all parameter neurons' weight and bias values based on the error values obtained between the final layer output neuron values and the actual data values. The error value is obtained from the predetermined cost function; Logistic binary cross entropy is used to measure the error value in the binary classification prediction because of its ability to provide a penalty in optimizing the error value. Logistic binary cross entropy provides a penalty against errors better than mean square error provides a penalty against errors in the optimization process.

$$C_{BCE}(a_j^L, C) = C\ log(1 + e^{-a_j^L}) + (1 - C)\ log(1 + e^{a_j^L}) \tag{8}$$

$$\frac{\delta}{\delta a_j^L} C_{BCE}(a_j^L, C) = -C\left(1 - \frac{e^{a_j^L}}{1+e^{a_j^L}}\right) + (1 - C)\frac{e^{a_j^L}}{1+e^{a_j^L}} = \frac{e^{a_j^L}}{1+e^{a_j^L}} - C \tag{9}$$

C_{BCE} : Cost function logistic binary cross entropy
C : Label class

The enhancement of error calculation precision for predicted-versus-actual data values, as derived from (9), involves adjusting weight and bias parameters across iterations. This adjustment aims to attain a global minimum [27]. Within the backward propagation process, stochastic gradient descent is employed. This technique modifies input data into diverse batches during each iteration. This approach serves two purposes, which is diminishing the computation time essential for training and conserving memory allocation [28].

Stochastic gradient descent computes the error by evaluating the partial derivative of the loss function expressed in (8) for every data point within the data batch, as expressed in (9). The optimization process involves adjusting the weight values for each neuron in each layer through (10), while the bias values are updated via (11). This weight and bias adjustment occurs using (12) and (13), wherein a predefined learning rate is incorporated. This iterative procedure strives for a global minimum, where minimal error is attained, repeating these steps across each iteration.

$$\frac{\delta C_{BCE}(a_j^L,C)}{\delta w_{j,k}^L} = \frac{\delta C_{BCE}(a_j^L,C)}{\delta a_j^L}\frac{\delta a_j^L}{\delta z_j^L}\frac{\delta z_j^L}{\delta w_{j,k}^L} \tag{10}$$

$$\frac{\delta C_{BCE}(a_j^L,C)}{\delta b_j^L} = \frac{\delta C_{BCE}(a_j^L,C)}{\delta a_j^L}\frac{\delta a_j^L}{\delta z_j^L}\frac{\delta z_j^L}{\delta b_j^L} \tag{11}$$

$$w_{(j,k)_{i+1}}^L = w_{(j,k)_i}^L - \alpha\frac{\delta C_{BCE}(a_j^L,C)}{\delta w_{j,k}^L} \tag{12}$$

$$b_{j_{i+1}}^L = b_{j_i}^L - \alpha\frac{\delta C_{BCE}(a_j^L,C)}{\delta b_j^L} \tag{13}$$

α : Learning Rate
i : Iteration

2.4. Performance evaluation

The methods used to evaluate the model's performance are precision, recall, accuracy, and F1. These four evaluation metrics are the most commonly used to evaluate model performance [29], the value of these four metrics is obtained from a comparison between the predicted value and the ground truth value. A true positive (TP) value is obtained if the predicted value has the same value as the ground truth positive value, a false positive (FP) is obtained if the model predicts a different value from the ground truth on positive ground truth, true negative (TN) is obtained if the model can predict Negative labels such as ground truth and false negative (FN) are obtained if the model predicts a different value from the ground truth on negative ground truth.

$$precission = \frac{TP}{TP+FP} \tag{14}$$

$$accuracy = \frac{TP+TN}{TP+FP+TN+FN} \tag{15}$$

$$recall = \frac{TP}{TP+FN} \tag{16}$$

$$F1 = \frac{2\times precission\times recall}{precission+recall} \tag{17}$$

3. RESULTS AND DISCUSSION

Pre-processing begins with splitting the dataset into training data (80%) and test data (20%) randomly from a dataset of 62,630 rows with 15 features with 939,450 data. The process of eliminating features that are not important is done to reduce the overfitting process of redundant features and the resulting elimination of two features. The resulting dataset is 814,190, with 651,352 training data and 50,104 test data. The data transformation process is carried out by scaling the data by converting the data into z-score form to maintain the scale of the data.

3.1. Feature selection

The selection of the ranking of the features is made using a random forest based on the gini impurity value of each feature, the results are obtained as represented using the errorbar graph in Figure 3. The mean decrease value is obtained from the average impurity value for each feature of all trees in the forest and the error value represents the variation of each gini impurity in each feature of all the trees in the forest. These data show that pressure, TVOC, and humidity have an important role in determining the amount of smoke produced in the fire detection. The presence of TVOC has an important role in detecting the presence of fire because it is based on readings of contents such as benzene, formaldehyde, and toluene in the air. This is strengthened by data from Vilčeková *et al.* [30], which shows that smoke presence is strongly influences by TVOC, PM2.5, and PM10 factors. The use of humidity sensors, barometric pressure sensors, and temperature sensors in fire detection systems has been used by several authors. It proves the correlation between the presence of fire and these parameters [6], [31].

The cross-validation process is carried out by selecting a value of k = 10 with the Random Forest Classifier as the model, and the optimal number of features reaches 7 features, with feature data Humidity, TVOC, H2, Ethanol, Pressure, PM1.0 and NC0.5. And the data that were eliminated were temperature, PM2.5, NC1.0, eCO2, and NC2.5 data. There is a difference between the important feature results and those obtained using simple random forest feature selection and recursive feature selection with RFECV. The difference is due to the difference in the method used. The RFECV method selects important features based on the relationship between features on prediction accuracy. In contrast, the random forest selects essential features based on the gini impurity value of each feature.

The outcomes derived from RFECV illustrate fluctuations in accuracy values for each employed feature selection during training. Throughout the training process, accuracy remains relatively consistent, with a value of 0.99, as depicted in Figure 4. However, adopting a single feature leads to a decline in accuracy, registering at 0.94. On the other hand, utilizing two features counteracts this drop and increases accuracy by 0.98, which matches that of using a single feature. For heightened accuracy, the training procedure employs the number of features identified through RFECV, totaling seven features.

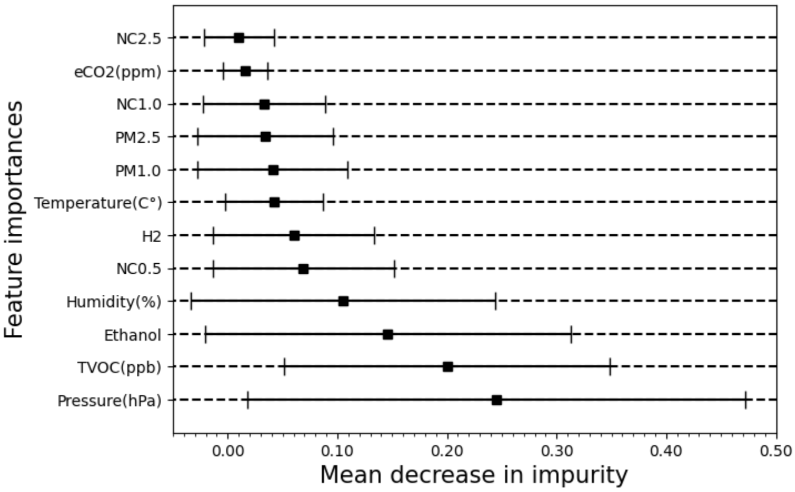


Figure 3. Mean decrease in impurity of dataset

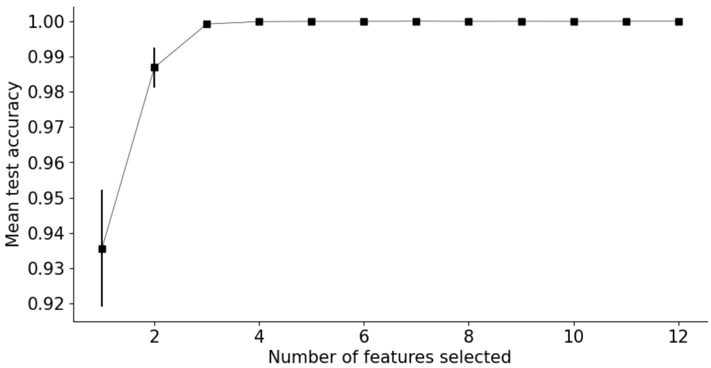


Figure 4. RFECV mean test accuracy toward feature eliminaions

3.2. Design system

The design of the system is shown in Figure 5 refers to the features that have been selected using RFECV; the feature is humidity (%). TVOC (ppb), ethanol, air pressure, PM1.0 and NC0.5. Selection of the main computing system using the ESP32 microcontroller as the main control and as a deep learning prediction processing device; the use of ESP32 as a microcontroller is due to its inference time in predicting 10,000 deep learning operations within 1599µs [32]. The choice of the BMP388 sensor is due to its accuracy in measuring pressure 300-1000hPa with an R2 value of 0.999 [33], SPS30 was chosen because it has an R2 of 0.98 on PM2.5 measurements and 0.98 on PM1.0 in indoor areas [34]. Humidity and temperature sensors using SHT31 have a more extensive range of humidity and more minor uncertainties than other sensors, such as DHT11 and DHT22 [35], and has higher accuracy and reading range for humidity and temperature parameters than BME688 describe in datasheet. The use of SGP41 as a sensor for measuring ethanol gas is used because of the accuracy and flexibility of the sensor in detecting volatile organic compound (VOC) gases. SGP41 has a wide range of VOC gases, one of which is acetone, acetonitrile, benzene, diethyl ether, ethanol, hexane, isopropanol, methanol, and toluene where these gases are an indication of a gas leak that can spread to a fire. SGP41 can measure these gases with a fast step response time, low concentrations, and varied gas distances than the BME688 [36]. The sensors are selected based on the needs, accuracy, and range of the type of parameter measurement being detected.

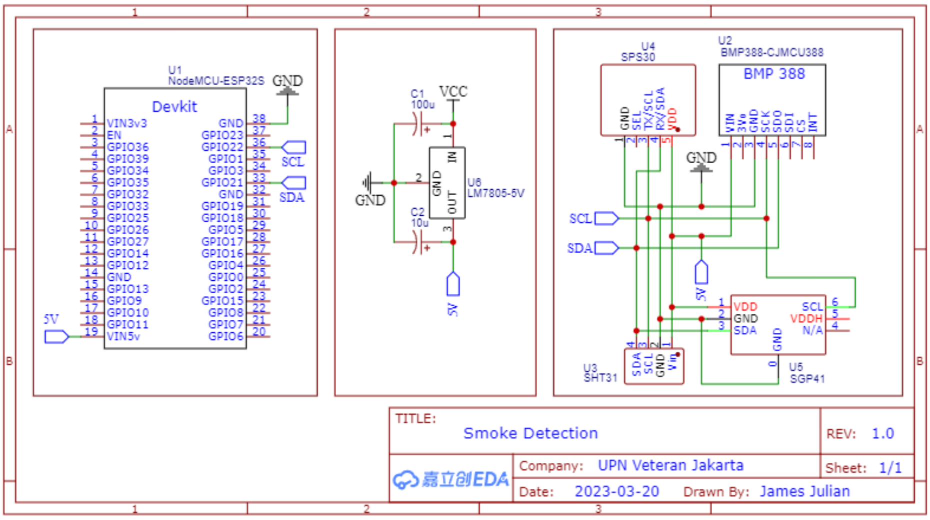


Figure 5. Schematic circuit of deep learning-based smoke detector

3.3. Training

The architectural model consists of 3 layers, one input layer, one hidden layer, and one input layer. The input layer consists of 12 neurons, each representing the input value of each parameter in the dataset, which is then continued to the hidden layer after calculating each weight and bias with a linear activation function. After the calculations for each node have been carried out, then proceed to the hidden layer using the ReLU activation function, which consists of 16 nodes, then a similar analysis is carried out to the output layer with the number of nodes 1 to state whether smoke is detected or not, the sigmoid function activation is used in binary classification with the logistic binary cross entropy function as a cost function with stochastic gradient descent as parameter optimization in the backpropagation process. The parameters of the architecture of each layer are shown in Table 1.

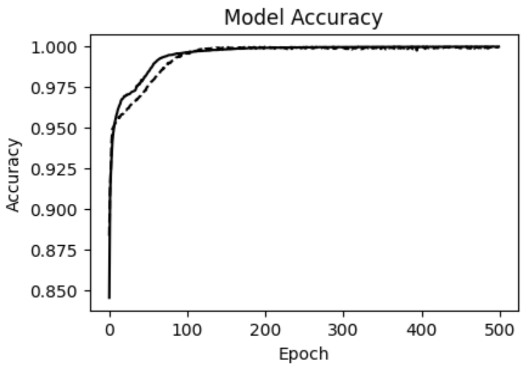
Table 1. Architecture summary

Layer Type	Node	Parameter
Dense	12	156
Dense	16	208
Dense	1	17
Total Parameters		381

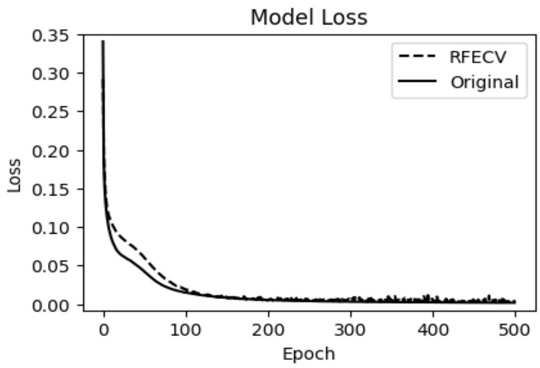
The training process is carried out with 500 iterations. The results obtained from the training accuracy and loss graphs are shown in Figure 6(a) by comparing the training process carried out using 12 features and training carried out using seven features that have been obtained in the feature selection process using RFECV. It was found that the two features have relatively the same accuracy, namely 0.99. The training process using

RFECV achieves faster optimization than the training process using 12 features; this is because the fewer features used, the fewer parameters are actively involved, so the optimization process in RFECV training is higher than the original using 12 features. Moreover, the loss graph as shown in Figure 6(b) RFECV has a smaller error value at the initial iteration stage than the original; this is because in the backward propagation process the number of parameters affects the complexity of the model, so the smaller the number of features given, the lower the level of complexity of the model [37].

The trained model is converted into a Tensor Flow Lite (TFLite) model, where the weight, activation, and bias are quantized to reduce memory size and computation time. The model from TFLite was then converted into the C++ model to be tested on the ESP32 microcontroller using standardized test data. Data testing is done by testing 1,000 samples of test data a thousand times to find out the performance of the converted model using (14)-(17) and the latency time of each prediction before and after the RFECV process is carried out.



(a)



(b)

Figure 6. Deep learning training result over iteration; (a) accuracy over 500 iteration and (b) loss over 500 iteration

The outcomes of the performance assessment are shown in Table 2. The performance results from training the initial model with 12 features and the RFECV-selected seven features exhibit comparably similar outcomes. Notably, the RFECV process showcases an average latency time that is notably faster, approximately 71.76% quicker than the original model. This translates to a latency time of 34.02 µs for the RFECV process, featuring a standard deviation 0.013. The original model exhibits a latency time of 47.41 µs with a standard deviation of 0.015.

Table 2. Performance evaluation model on ESP32

Model	TP	FP	TN	FN	Accuracy	Precission	Recall	F1-score	Latency (*µ*s)
Original	716	2	281	1	1.00	1.00	1.00	1.00	47.41
RFECV	717	10	273	0	0.99	0.99	1.00	0.99	34.02

4. CONCLUSION

Through the research and literature studies that have been carried out as shown in section 3, it is known that this deep learning algorithm-based smoke detection system has an accuracy of 0.99 with a latency time of prediction of 34.02 µs, 71.76% faster than prediction performance using 12 parameters before the RFECV process which is has been tested on ESP32 microcontroller. This proves that using RFECV can increase the effectiveness of predictions without significantly decreasing accuracy. The selection of sensors in the design process based on the RFECV process results in the elimination of 5 features from 12 original features in the dataset; the features used in the training process are moisture, TVOC, pressure, H2, ethanol, PM10, and NC0.5. Based on the results of the selection of the sensor based on the RFECV result parameters, the sensors are SHT31 to read humidity values, SGP41 to read ethanol, SPS30 to read the PM1.0 and NC0.5, and BMP388 to read air pressure; ESP32 microcontroller chosen as a central processing system to predict whereabouts as soon as possible based on sensor readings. In the future, it is necessary to carry out experimental research to prove the accuracy of the predictions from the model and prove the accuracy of the system design that has been made as a validation step.

REFERENCES

[1] M. McNamee *et al.*, "IAFSS agenda 2030 for a fire safe world," *Fire Safety Journal*, vol. 110, p. 102889, Dec. 2019, doi: 10.1016/j.firesaf.2019.102889.

[2] A. Hamins *et al.*, "Reducing the risk of fire in buildings and communities: A strategic roadmap to guide and prioritize research," pp. 37–46, 2013, doi: 10.3850/978-981-07-5936-0_01-05.

[3] J. Milke and R. Zevotek, "Analysis of the response of smoke detectors to smoldering fires and nuisance sources," *Fire Technology*, vol. 52, no. 5, pp. 1235–1253, Feb. 2016, doi: 10.1007/s10694-015-0465-2.

[4] B. Sun, P. Cheng, and Y. Huang, "Few-shot fine-grained forest fire smoke recognition based on metric learning," *Sensors*, vol. 22, no. 21, p. 8383, Nov. 2022, doi: 10.3390/s22218383.

[5] A. Jadon, M. Omama, A. Varshney, M. S. Ansari, and R. Sharma, "Firenet: A specialized lightweight fire and smoke detection model for real-time iot applications," 2019, [Online]. Available: http://arxiv.org/abs/1905.11922.

[6] D. M. Doolin and N. Sitar, "Wireless sensors for wildfire monitoring," in *Smart Structures and Materials 2005: Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems*, May 2005, vol. 5765, p. 477, doi: 10.1117/12.605655.

[7] A. L. Holder *et al.*, "Field evaluation of low-cost particulate matter sensors for measuring wildfire smoke," *Sensors (Switzerland)*, vol. 20, no. 17, pp. 1–17, Aug. 2020, doi: 10.3390/s20174796.

[8] S. J. Chen, D. C. Hovde, K. A. Peterson, and A. W. Marshall, "Fire detection using smoke and gas sensors," *Fire Safety Journal*, vol. 42, no. 8, pp. 507–515, Nov. 2007, doi: 10.1016/j.firesaf.2007.01.006.

[9] A. Solórzano *et al.*, "Early fire detection based on gas sensor arrays: Multivariate calibration and validation," *Sensors and Actuators B: Chemical*, vol. 352, p. 130961, Feb. 2022, doi: 10.1016/j.snb.2021.130961.

[10] V. T. Pham, Q. B. Le, D. A. Nguyen, N. D. Dang, H. T. Huynh, and D. T. Tran, "Multi-sensor data fusion in a real-time support system for on-duty firefighters," *Sensors (Switzerland)*, vol. 19, no. 21, p. 4746, Nov. 2019, doi: 10.3390/s19214746.

[11] H. Alqourabah, A. Muneer, and S. M. Fati, "A smart fire detection system using IoT technology with automatic water sprinkler," *International Journal of Electrical and Computer Engineering*, vol. 11, no. 4, pp. 2994–3002, Aug. 2021, doi: 10.11591/ijece.v11i4.pp2994-3002.

[12] A. H. Miry and G. A. Aramice, "Water monitoring and analytic based ThingSpeak," *International Journal of Electrical and Computer Engineering*, vol. 10, no. 4, pp. 3588–3595, Aug. 2020, doi: 10.11591/ijece.v10i4.pp3588-3595.

[13] R. Teguh, F. F. Adji, Benius, and M. N. Aulia, "Android mobile application for wildfire reporting and monitoring," *Bulletin of Electrical Engineering and Informatics*, vol. 10, no. 6, pp. 3412–3421, Dec. 2021, doi: 10.11591/eei.v10i6.3256.

[14] A. L. Jutinico, G. A. R. Rodriguez, and J. R. C. Lopez, "Wearable sensor network for lower limb angle estimation in robotics applications," *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 21, no. 2, pp. 390–399, Apr. 2023, doi: 10.12928/TELKOMNIKA.v21i2.23483.

[15] M. J. Fadhil, R. A. Fayadh, and M. K. Wali, "Design and implementation a prototype system for fusion image by using SWT-PCA algorithm with FPGA technique," *International Journal of Electrical and Computer Engineering*, vol. 10, no. 1, pp. 757–766, Feb. 2020, doi: 10.11591/ijece.v10i1.pp757-766.

[16] S. Blattmann, "Smoke detection dataset," 2022, [Online]. Available: https://www.kaggle.com/datasets/deepcontractor/smoke-detection-dataset.

[17] O. Bayasli and H. Salhi, "The cubic root unscented kalman filter to estimate the position and orientation of mobile robot trajectory," *International Journal of Electrical and Computer Engineering*, vol. 10, no. 5, pp. 5243–5250, Oct. 2020, doi: 10.11591/IJECE.V10I5.PP5243-5250.

[18] S. Parvandeh, H. W. Yeh, M. P. Paulus, and B. A. McKinney, "Consensus features nested cross-validation," *Bioinformatics*, vol. 36, no. 10, pp. 3093–3098, Jan. 2020, doi: 10.1093/bioinformatics/btaa046.

[19] T. A. Assegie, P. K. Rangarajan, N. K. Kumar, and D. Vigneswari, "An empirical study on machine learning algorithms for heart disease prediction," *IAES International Journal of Artificial Intelligence*, vol. 11, no. 3, pp. 1066–1073, Sep. 2022, doi: 10.11591/ijai.v11.i3.pp1066-1073.

[20] S. Feng, H. Zhou, and H. Dong, "Using deep neural network with small dataset to predict material defects," *Materials and Design*, vol. 162, pp. 300–310, Jan. 2019, doi: 10.1016/j.matdes.2018.11.060.

[21] N. Yuvaraj, R. A. Raja, N. V. Kousik, P. Johri, and M. J. Diván, "Analysis on the prediction of central line-associated bloodstream infections (CLABSI) using deep neural network classification," in *Computational Intelligence and Its Applications in Healthcare*, Elsevier, 2020, pp. 229–244.

[22] A. Zhang, Z. C. Lipton, M. Li, and A. J. Smola, "Dive into deep learning," *Journal of the American College of Radiology*, vol. 17, no. 5, pp. 637–638, Jun. 2021, [Online]. Available: https://doi.org/10.1016%2Fj.jacr.2020.02.005.

[23] O. A. Montesinos López, A. Montesinos López, and J. Crossa, "Fundamentals of artificial neural networks and deep learning," in *Multivariate Statistical Machine Learning Methods for Genomic Prediction*, Springer International Publishing, 2022, pp. 379–425.

[24] A. Nguyen, K. Pham, D. Ngo, T. Ngo, and L. Pham, "An analysis of state-of-the-art activation functions for supervised deep neural network," in *Proceedings of 2021 International Conference on System Science and Engineering, ICSSE 2021*, Aug. 2021, pp. 215–220, doi: 10.1109/ICSSE52999.2021.9538437.

[25] M. R. Zadeh, S. Amin, D. Khalili, and V. P. Singh, "Daily outflow prediction by multi layer perceptron with logistic sigmoid and tangent sigmoid activation functions," *Water Resources Management*, vol. 24, no. 11, pp. 2673–2688, Mar. 2010, doi: 10.1007/s11269-009-9573-4.

[26] J. C. R. Whittington and R. Bogacz, "Theories of error back-propagation in the brain," *Trends in Cognitive Sciences*, vol. 23, no. 3, pp. 235–250, Mar. 2019, doi: 10.1016/j.tics.2018.12.005.

[27] V. Andrearczyk and P. F. Whelan, "Deep learning in texture analysis and its application to tissue image classification," in *Biomedical Texture Analysis: Fundamentals, Tools and Challenges*, Elsevier, 2017, pp. 95–129.

[28] T. Vatanen, T. Raiko, H. Valpola, and Y. LeCun, "Pushing stochastic gradient towards second-order methods - Backpropagation learning with transformations in nonlinearities," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 8226 LNCS, no. PART 1, Springer Berlin Heidelberg, 2013, pp. 442–449.

[29] B. J. Erickson and F. Kitamura, "Magician's corner: 9. performance metrics for machine learning models," *Radiology: Artificial Intelligence*, vol. 3, no. 3, p. e200126, May 2021, doi: 10.1148/ryai.2021200126.

[30] S. Vilčeková, I. Z. Apostoloski, Ľ. Mečiarová, E. K. Burdová, and J. Kiseľák, "Investigation of indoor air quality in houses of Macedonia," *International Journal of Environmental Research and Public Health*, vol. 14, no. 1, p. 37, Jan. 2017, doi: 10.3390/ijerph14010037.

[31] Y. E. Aslan, I. Korpeoglu, and özgür Ulusoy, "A framework for use of wireless sensor networks in forest fire detection and

monitoring," *Computers, Environment and Urban Systems*, vol. 36, no. 6, pp. 614–625, Nov. 2012, doi: 10.1016/j.compenvurbsys.2012.03.002.
[32] M. Z. H. Zim, "TinyML: Analysis of Xtensa LX6 microprocessor for neural network applications by ESP32 SoC," 2021, doi: 10.13140/RG.2.2.28602.11204.
[33] B. Ganev, D. Nikolov, and M. B. Marinov, "Performance evaluation of MEMS pressure sensors," Jul. 2020, doi: 10.1109/ELECTRONICA50406.2020.9305140.
[34] I. Patwardhan, S. Sara, and S. Chaudhari, "Comparative evaluation of new low-cost particulate matter sensors," in *Proceedings - 2021 International Conference on Future Internet of Things and Cloud, FiCloud 2021*, Aug. 2021, pp. 192–197, doi: 10.1109/FiCloud49777.2021.00035.
[35] Z. Shao, M. Huang, D. Wu, X. Zhang, and A. Huang, "Design of a simplified wireless sensor network node based on MQTT protocol," *DEStech Transactions on Computer Science and Engineering*, no. iece, Dec. 2018, doi: 10.12783/dtcse/iece2018/26633.
[36] M. F. R. Al-Okby, T. Roddelkopf, H. Fleischer, and K. Thurow, "Evaluating a novel gas sensor for ambient monitoring in automated life science laboratories," *Sensors*, vol. 22, no. 21, p. 8161, Oct. 2022, doi: 10.3390/s22218161.
[37] X. Ying, "An overview of overfitting and its solutions," *Journal of Physics: Conference Series*, vol. 1168, no. 2, p. 22022, Feb. 2019, doi: 10.1088/1742-6596/1168/2/022022.

BIOGRAPHIES OF AUTHORS

James Julian was born in Jakarta on July 18th, 1986. He received his bachelor's degree in mechanical engineering form Universitas Pancasila in 2012 and obtained his master's degree form Universitas Indonesia in 2015. Futhermore, he received his Doctor of Engineering degree also form Univeristas Indonesia. Currently he serves as Lecture in Department of Mechanical Engineering, Universitas Pembangunan Nasional Veteran Jakarta (UPNVJ), Indonesia. His research is currently focused on fluid mechanics, machine learning and especially on the application of a plasma actuator on flow control. He is also interested in the application of computational fluid dynamics. He can be contacted at email: zames@upnvj.ac.id.

Annastya Bagas Dewantara received the bachelor's degree in electrical engineering from the University Pembangunan Nasional Veteran Jakarta, Jakarta, Indonesian, was born in Jakarta on December 08th, 2001. His field includes electronics, microcontrollers, and control system. He is salso interested in embedded system, IoT and sensor network. He can be contacted at email: annastya.bd@upnvj.ac.id.

Fitri Wahyuni was born in Bukit Tinggi on July 2nd, 1985. She completed her Bachelor of Physics education at the Jakarta State University in 2009 and her master's in mechanical engineering at Gadjah Mada University in 2013. From 2014 to 2019 she worked as a Lecturer in Industrial Engineering, Indraprasta University, PGRI. From 2019 until now he has worked as a Lecturer in Mechanical Engineering at the Veterans National Development University in Jakarta. At present, the area of research that is being developed is Construction Machinery. She can be contacted at email: fitriwahyuni@upnvj.ac.id.