



## A New Approach for Detection of Viral Respiratory Infections Using E-nose Through Sweat from Armpit with Fully Connected Deep Network

**Malikhah Malikhah<sup>1</sup>**      **Riyanarto Sarno<sup>1\*</sup>**      **Sozo Inoue<sup>2</sup>**      **M. Syauqi Hanif Ardani<sup>1</sup>**  
**Doni Putra Purbawa<sup>1</sup>**      **Shoffi Izza Sabilla<sup>1</sup>**      **Kelly Rossa Sungkono<sup>1</sup>**      **Chastine Fatichah<sup>1</sup>**  
**Dwi Sunaryono<sup>1</sup>**      **Arief Bakhtiar<sup>3</sup>**      **Libriansyah<sup>4</sup>**      **Cita RS. Prakoeswa<sup>5</sup>**  
**Damayanti Tinduh<sup>6</sup>**      **Yetti Hernaningsih<sup>7</sup>**

<sup>1</sup> *Department of Informatics, Faculty of Intelligent Electrical and Informatics Technology, Institut Teknologi Sepuluh Nopember (ITS) Sukolilo, Surabaya, Indonesia*

<sup>2</sup> *Department of Human Intelligence Systems, Kyushu Institute of Technology Kitakyushu, Japan*

<sup>3</sup> *Department of Pulmonology, Faculty of Medicine, Airlangga University, Surabaya, Indonesia*

<sup>4</sup> *Department of Internal Medicine, Dr. Ramelan Navy Hospital Surabaya, Indonesia*

<sup>5</sup> *Department of Dermatology and Venerology, Faculty of Medicine, Airlangga University, Surabaya, Indonesia*

<sup>6</sup> *Department of Physical Medical and Rehabilitation, Faculty of Medicine, Airlangga University, Surabaya, Indonesia*

<sup>7</sup> *Department of Clinical Pathology, Faculty of Medicine, Airlangga University, Surabaya, Indonesia*

\* Corresponding author's Email: [riyanarto@if.its.ac.id](mailto:riyanarto@if.its.ac.id)

---

**Abstract:** Viral respiratory infections are the most common diseases suffered by all age groups worldwide. The gold standard for diagnosing viral respiratory infection is through the molecular method, but this diagnosis is expensive, requires sophisticated equipment, can only be performed by well-trained medical staff, and is painful. Volatile Organic Compounds (VOCs) are compounds released from the human body that can be a marker of disease and based on numerous studies it also contains VOCs. An electronic nose (E-nose) is a device that can be used to identify disease. This study proposes a new approach for the detection of viral respiratory infections through sweat from the armpit using an E-nose consisting of 5 semiconductor gases and a single-board computer. Several statistical parameters are used to obtain features and the detection algorithm used is Fully Connected Deep Network (FCDN). Several sizes of hidden layers were tested to obtain the best FCDN model. This study also proposes the selection of the best FCDN model which is a trade-off between complexity and accuracy, so that the model stored in E-nose is a model that not only has good accuracy but is also not too complex. The experimental results show that using 29 statistical parameters and 2 hidden layers generate the highest accuracy of 0.940 for the detection of 2 classes, namely positive and negative, with sensitivity and specificity of 0.967 and 0.915, respectively, where the best FCDN model has a total of 90,561 parameters.

**Keywords:** Armpit, Electronic nose, Deep learning, Sweat, Statistical parameters, Viral respiratory infections.

---

### 1. Introduction

Viral respiratory infections are the most common diseases suffered by all age groups worldwide. Elderly people and children have the highest risk of being infected with the disease, especially in low-income countries. A particular report states that more than 16 % of deaths occur in children under 5 years old [1]. The spread of this disease is rapid, through

direct physical contact, droplets, or aerosols. Quick and accurate diagnosis is needed so that the subject receives the right treatment and minimizes the spread of the disease. The gold standard for diagnosing this disease is through the molecular method, but this diagnosis is expensive, requires sophisticated equipment, can only be performed by well-trained medical staff, and is painful [2].

Volatile organic compounds or better known by

the abbreviation VOCs are compounds that easily evaporate at a certain pressure and temperature. Hundreds of VOCs have been reported to be excreted by humans through breath, saliva, skin, sweat, blood, milk, urine, and feces [3]. VOCs released through sweat have been studied to determine whether or not sweat can be a signal of certain diseases and the results show that sweat can be a potential sample to detect the presence of disease in the body [4]. In the human body, sweat is found all over the skin but is mostly produced on the soles of the feet, hands, armpits, and genital areas [5]. By using a sample from the armpit, the detection will be safer because the virus that causes viral respiratory infections is not found in sweat [6].

An electronic nose (E-nose) is an electronic sensing device that works similarly to a human nose to detect odors [7]. E-nose has been used in various fields, such as cosmetic, pharmaceutical, food and beverage, automobile, agriculture, and medical [8]. In the medical field, the use of e-nose nowadays increases to identify pathogens, select the right treatment for patients, identify nutritional status, psychological conditions, and detect diseases [9]. An electronic nose is composed of several sensors and a data processing unit. In the data processing unit, there is a classification algorithm that can be used to detect diseases. The type of sensor that is most widely used in E-nose is Metal-oxide Sensor (MOS) because it is highly sensitive, inexpensive, and fast response.

Studies using E-nose to detect diseases have been conducted. Previous research has been successfully detected diseases caused by *aspergillus fumigatus* through exhaled breath using canonical discriminant analysis (CDA) with an accuracy of 89 % [10]. Another research in 2016 has been designed e-nose to detect joint inflammatory through exhaled breath disease using principal component analysis (PCA) and linear discriminant analysis (LDA) with an accuracy of 70 % [11]. Meanwhile, Saidi et. al has created an e-nose device to distinguish chronic kidney disease, diabetes mellitus, and healthy subjects through exhaled breath using PCA and support vector machine (SVM) with perfect accuracy [12]. Another research in 2018 by Yang et. al tried to detect pneumoconiosis through exhaled breath using LDA with an accuracy of 70 % [13]. However, no previous research has used sweat from the armpit as an E-nose sample to detect disease. The latest research using e-nose to detect diabetes through urine using deep learning with convolutional neural network (CNN) produces an accuracy of 100 % [14]. The advantages of deep learning compared to other detection algorithms are that it provides quality final results, can manipulate data more effectively and can

handle data signals [15]. However, CNN is famous for its high complexity, so it is not suitable to be applied to an e-nose that is built using a single-board computer. In other areas, another type of deep learning namely fully connected deep network (FCDN) has been used to identify the type of meat and the results prove that FCDN is also able to help e-nose to detect with lower complexity but comparable accuracy [16].

Statistical parameters have been used in E-nose to extract features, including using mean, standard deviation, minimum, and maximum for the detection of civet and non-civet coffee [17, 18], furthermore mean, standard deviation, skewness, and kurtosis also used for the detection of mixed beef and pork [19, 20]. Other statistical parameters, such as variance and root mean square has been also used in audio data [21], but their use is still limited to E-nose signal data.

As mentioned before, the self-developed e-nose is designed with a single-board computer, so that if needed to store the detection algorithm in E-nose, FCDN model must not be too complex, but must also have reliable accuracy, at the same time [22]. Complexity in FCDN can be represented by the network depth and width of each layer or the number of parameters [23]. Accuracy alongside and numbers of parameters in FCDN must be taken into the account in determining the selection of models to be stored in E-nose from several FCDN models that have been built.

The use of an E-nose that takes samples of sweat from the armpit, simple statistical parameter feature extraction, and a less complex but accurate FCDN model which is automatically selected from several FCDN models as a detection algorithm are the main advantages of this research compared to others. Moreover, this research aims that the self-developed e-nose can help health workers carry out initial screening for the detection of viral respiratory infections quickly, accurately, painlessly, non-invasively, inexpensively, and not increase the risk of transmitting viral respiratory infections.

Therefore, the contributions to this study are: (i) a new approach to detect viral respiratory infections from sweat in the armpit using E-nose; (ii) addition of statistical parameter feature extractions from previous studies; and (iii) automatic selection of the best model using an indicator which is a trade-off between complexity and accuracy.

This paper is structured as follows: Section 1 provides the background of our research. Section 2 covers the proposed method. Section 3 explains the experimental setup, result of experiments and

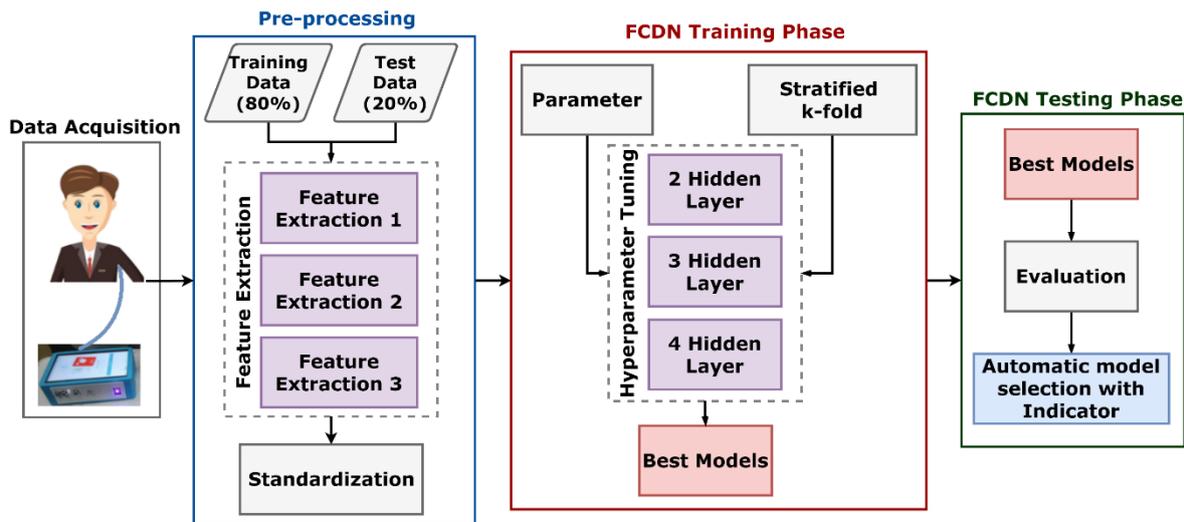


Figure. 1 The proposed method

discussion. Section 4 summarizes the main point of our research.

## 2. Proposed method

This study aims to build an E-nose that uses sweat from the armpit as the sample to detect viral respiratory infections. The proposed material and method are illustrated in Fig. 1.

### 2.1 Pre-processing

The data obtained from the data acquisition stage is divided into training data and test data of 80 % and 20 %, respectively, where the data would be further processed using statistical parameter feature extraction and standardization. The output of this stage would be input for FCDN training and testing phase.

#### 2.1.1. Statistical parameter feature extraction

The data signal generated by the E-nose is stationary, so statistical parameters can be calculated. This study used 29 statistical parameters, in which there are 3 combinations of statistical parameter feature extraction as can be seen in Table 1.

Percentile is the lowest value equal to or greater than  $p$  % of the value in the data. Eq. (1) is used to calculate percentile,  $p$  is the desired percentile, and  $x_t$  is the value of each observed data.

$$\text{Percentile}(p) = p\% \times x_t \quad (1)$$

Variance is the sum of the squares of all the deviations of the individual values with the data mean. Variance is calculated using Eq. (2), where  $x_t$  is the value of each observed data,  $\mu$  is the data average, and  $n$  is the number of observations.

Table 1. Combination of statistical parameter feature extraction

Feature extraction	Statistical parameters
Feature extraction 1	4 statistical parameters: Mean, standard deviation, minimum, maximum
Feature extraction 2	4 statistical parameters: Mean, standard deviation, skewness, kurtosis
Feature extraction 3	29 statistical parameters: Mean, standard deviation, minimum, maximum, skewness, kurtosis, 5 <sup>th</sup> percentile, 10 <sup>th</sup> percentile, 15 <sup>th</sup> percentile, 20 <sup>th</sup> percentile, 25 <sup>th</sup> percentile, 30 <sup>th</sup> percentile, 35 <sup>th</sup> percentile, 40 <sup>th</sup> percentile, 45 <sup>th</sup> percentile, 50 <sup>th</sup> percentile, 55 <sup>th</sup> percentile, 60 <sup>th</sup> percentile, 65 <sup>th</sup> percentile, 70 <sup>th</sup> percentile, 75 <sup>th</sup> percentile, 80 <sup>th</sup> percentile, 85 <sup>th</sup> percentile, 90 <sup>th</sup> percentile, 95 <sup>th</sup> percentile, varians, root mean square, zero crossing, and mean crossing.

$$S^2 = \frac{\sum(x_t - \mu)^2}{n} \quad (2)$$

Root mean square (RMS) is the root of the average value of a function squared. RMS is calculated using Eq. (3), where  $x_n$  is the value of each observed data, and  $n$  is the number of observations.

$$\text{RMS} = \sqrt{\frac{1}{n} \sum_n x_n^2} = \sqrt{\frac{x_1^2 + \dots + x_n^2}{n}} \quad (3)$$

Zero-crossing (ZC) indicates the number of times adjacent signals pass through the horizontal axis which is 0. ZC is calculated using Eq. (4), where  $T$  is

the number of observations, and  $x(t)$  is the value of each observed data.

$$ZC = \frac{1}{2} \sum_{t=1}^T |x(t) - x(t-1)| \quad (4)$$

Mean crossing (MC) indicates the number of times the signals of two adjacent values pass through the mean. MC is calculated using Eq. (5), where  $T$  is the number of observations,  $\mu$  is the data average, and  $x(t)$  is the value of each observed data.

$$MC = \frac{\sum_{t=2}^T |(x(t)-\mu)-(x(t-1)-\mu)|}{2} \quad (5)$$

### 2.1.2. Standardization

Data standardization is needed in a quantitative variable study that has different units. It is used in this study so that the data scale is not too different; this is intended so that the average of the observed data becomes 0 and the standard deviation becomes 1. The standard used in this study is the standard scaler in python [24]. Eq. (6) is used to standardize the data, where  $z$  is the new data,  $x$  is the observed data,  $\mu$  is the average of all data, and  $\sigma$  is the standard deviation of the data.

$$z = \frac{x-\mu}{\sigma} \quad (6)$$

### 2.2 Fully connected deep network training phase

Fully connected deep network (FCDN) is one of the most commonly used types of deep learning. The difference between FCDN and artificial neural network (ANN) is in the number of hidden layers, where FCDN has more than 1 hidden layer [16]. FCDN consists of a fully connected layer where all active neurons from the previous layer are connected to neurons in the next layer. In the detection of viral respiratory infections, the number of input layers from the FCDN corresponds to the number of features obtained from the statistical parameter feature extraction. The input layer then connects it to the first hidden layer, and the first hidden layer

connects to the next hidden layer. The last hidden layer will pass it to the output layer. Each neuron receives input and performs a dot operation with weight, a parameter value that connects one neuron to another.

In FCDN training phase, this study performs hyperparameter tuning to determine the best number of neurons that make up FCDN. Several hidden layers were tested, namely 2, 3, and 4 hidden layers. The number of neurons searched for each hidden layer is shown in Table 2. Hyperparameter tuning is conducted using random search, where not all values are tested; the values tested are random and can be limited as desired [25]. Due to limited resources, this study used random search with 50 iterations.

As explained earlier, each neuron connected to other neurons is connected by a parameter called weight, where the value of this weight will be different for each layer. To determine the  $i$ -th output of the fully connected layer, Eq. (7) is used, where  $\sigma$  is the activation function,  $w_m$  is the  $m$ -th weight, while  $x$  is the input of the fully connected layer.

$$y_i = \sigma (w_1 x_1 + \dots + w_m x_m) \quad (7)$$

Each layer consists of neurons and an activation function that determines whether a neuron is active or not. This study uses ReLU as an activation function that connects the hidden layer because ReLU is an activation function that generates better results in a number of experiments [26]. ReLU is calculated using Eq. (8), in which it will make a delimiter on the number zero, meaning that if  $x \leq 0$  then  $x = 0$  and if  $x > 0$  then  $x = x$ .

$$\frac{f(y)}{f(x)} = \begin{cases} x & x > 0 \\ 0 & x \leq 0 \end{cases} \quad (8)$$

Meanwhile, the activation function at the output layer uses a sigmoid calculated using Eq. (9) because this study solves binary problems.

$$f(x) = \frac{1}{1+e^{-x}} \quad (9)$$

Table 2. Hyperparameter ranges of FCDN used for the experiment in different number of hidden layer

Hidden Layer	Hyperparameter	Choice
2 Hidden Layer	Number of neurons in the 1 <sup>st</sup> hidden layer	range(64, 256)
	Number of neurons in the 2 <sup>nd</sup> the hidden layer	range(256, 1024)
3 Hidden Layer	Number of neurons in the 1 <sup>st</sup> hidden layer	range(64, 256)
	Number of neurons in the 2 <sup>nd</sup> the hidden layer	range(256, 1024)
	Number of neurons in the 3 <sup>rd</sup> the hidden layer	range(256, 1024)
4 Hidden Layer	Number of neurons in the 1 <sup>st</sup> hidden layer	range(64, 256)
	Number of neurons in the 2 <sup>nd</sup> the hidden layer	range(256, 1024)
	Number of neurons in the 3 <sup>rd</sup> the hidden layer	range(64, 256)
	Number of neurons in the 4 <sup>th</sup> hidden layer	range(256, 1024)

When the learning model generates a high loss value, FCDN executes a function called a loss function, where errors can be minimized using this function. The loss function will compare the prediction of the output and the target using Eq. (10), where  $T$  is the target and  $P$  is the prediction. If the prediction is far from the target, the resulting model performance is not optimal.

$$\text{LossFunction} = (T - P)^2 \quad (10)$$

Each error in the loss function indicates that the weight and bias values must be adjusted, which is called back-propagation. There is also a parameter called optimizer, an algorithm to reduce the loss function and change the weights in backpropagation. Eq. (11) is used by the optimizer to change the weight with the learning rate value, where  $*W$  is a new weight,  $W_x$  is the old weight,  $a$  is the learning rate, and  $(\frac{\partial \text{error}}{\partial W_x})$  is derivative of error related to weight. This study uses Adam as an optimizer to train and test the model because it has advantages, such as requiring small memory, working well with large data, and parameters with efficient computation [27]. The learning rate used in this study is 0.0001.

$$*W_x = W_x - a \left( \frac{\partial \text{error}}{\partial W_x} \right) \quad (11)$$

Before creating a model, FCDN has several parameters that must be adjusted, namely epoch and batch size. This study uses 100 epochs and batch\_size 32. This study tried several epochs but increasing and decreasing epochs do not show better results.

### 2.3 Model selection using indicator

This study selects the best FCDN model from several FCDN models using an indicator that is a trade-off of accuracy and complexity automatically. The model to be selected is a model of which parameters are not too many but produces competitive accuracy; therefore, this study proposes the best model selection indicator using Eq. (12), where the model with the highest indicator value will be selected as the model to be stored in the E-nose device, where  $AccTest$  is the accuracy in the test data, and  $Param$  is the number of parameters in deep learning. The number of parameters is normalized using a logarithm with an Euler number basis or called the natural logarithm (ln) [28], because the number of deep learning parameters can be hundreds, thousands, even tens of millions. The goal of normalization is to convert a numerical value using a common scale, without distorting differences in the

range of values or losing information [29]. Algorithm 1 is pseudocode for selecting the best model automatically using indicators, where in the first step the program obtains accuracies, number of parameters and calculates the indicator (I) values for all FCDN models, and in the second step, the program chooses an FCDN model which has the highest indicator value.

$$I = (\text{Acc Test}) \times 100 - \ln(\text{Param}) \quad (12)$$

### 2.4 Evaluation

Cross-validation stratified k-fold is used in the FCDN training phase by using training data to select the best FCDN model from hyperparameter tuning. This study uses a stratified k-fold with 5 folds. Stratified k-fold with 5 folds means that the data were divided into 5 equal folds, where the class distribution is balanced according to the class distribution of the entire population in each fold.

FCDN models created in the training phase were then evaluated using data that were never used in the training phase, namely test data. The evaluation was done using a confusion matrix that consists of true positive (TP), true negative (TN), false positive (FP), and false negative (FN). TP means that the model correctly identifies the subject as having the disease, FP means that the model incorrectly identifies the subject as having the disease, even though the subject

---

#### Algorithm 1 Automatic model selection with indicator

---

**Input** : 9 Models from Hyperparameter Tuning  $M_T$   
Test Data  $D_T$

**Output** : *SelectedModel*

**Step 1:** obtain Accuracy, Number of parameters, indicator value of  $M_T$

$Acc, numParam, indicator = [], []$

**for**  $t = 1$  to  $T$  **do**

$Acc_t \leftarrow$  predict  $D_T$  with  $M_t$

$numParam_t \leftarrow$  calculate the number of parameters of  $M_T$

$indicator_t \leftarrow$  calculate indicator value using Eq. (12)

**end for**

**Step 2:** Select model which has the highest indicator value

$index \leftarrow$  get the index of  $\max(indicator)$

$SelectedModel = M_{index}$

**return** *SelectedModel*

---

does not have the disease, TN means that the model correctly identifies the subject as not having the disease, while FN means that the model incorrectly identifies the subject as not having the disease, when in fact the subject has the disease.

In the diagnostic and detection performance metrics, this study focuses on accuracy, sensitivity, and specificity. Accuracy is a measure of how well the model correlates the predicted results with the actual results. Sensitivity is the probability that a positive subject is identified as positive by the test; this indicates the ability of the model to detect a sick subject. Specificity is the probability that a negative subject is correctly identified as negative by the model; this indicates the ability of the model to detect a negative or healthy subject. To calculate accuracy, sensitivity, and specificity, Eqs. (13), (14), and (15) are used.

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \tag{13}$$

$$\text{Sensitivity} = \frac{TP}{TP+FN} \tag{14}$$

$$\text{Specificity} = \frac{TN}{TN+FP} \tag{15}$$

### 3. Results and discussion

#### 3.1 Experimental setup

This study develops an electronic nose consisting of 5 pre-selected metal-oxide sensors with code S1, S2, S3, S4, and S5, a single-board computer, and also an SHT sensor to monitor humidity and temperature in the sensor chamber. Metal-oxide sensors are assembled to arduino microcontroller. In addition, the E-nose also has a monitor used to view the data retrieval process.

There are three processes carried out by E-nose every time a sample is taken [16]. The first process is flushing, inhaling the air around the E-nose; the second process is the sampling process, where the E-nose takes sweat data from the armpit; meanwhile, the last process is cleaning, cleansing the E-nose so that it is ready to use for further sampling. The time required to obtain data is 190 seconds, where the first process takes 10 seconds, the second process takes 120 seconds, and the third process takes 60 seconds. In the sampling process, metal-oxide sensors on the E-nose inhale the volatile gases from sweat in the armpit through a hose for 120 seconds and process it into a digital signal using an analog to digital converter (ADC) and store it to a single-board

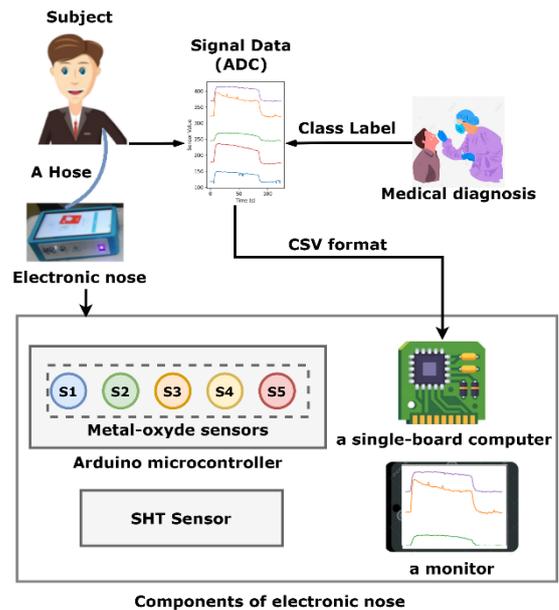


Figure. 2 Experimental setup

computer in the comma separated value (CSV) format. Each data obtained is also accompanied by a class label derived from medical diagnoses by health workers. The 2 class labels used are positive and negative, where the positive class means the subject has viral respiratory infections, while the negative class means the subject is healthy. A detailed description of the experimental setup for data acquisition is illustrated in Fig. 2.

According to the declaration of the helsinki by world medical association (WHO), all studies involving humans as research subjects are required to obtain the consent of the subject. This study has informed all subjects about the purposes, methods,

Table 3. Data distributions

Class	Total	Training data	Test data
Negative	353	282	71
Positive	306	245	61
All	659	527	132

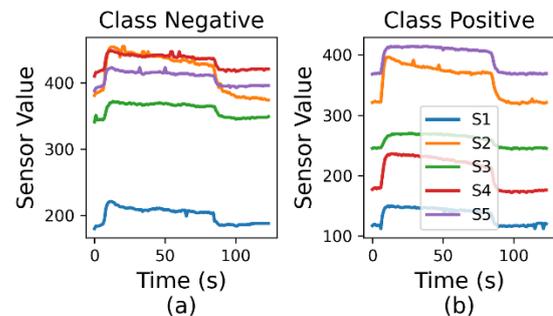


Figure. 3 examples of signal obtained by e-nose: (a) negative class and (b) positive class

and medical procedures; and all subjects have signed the medical consent form.

### 3.2 Results of data acquisition

Data acquisition was carried out for 3 months and the total data obtained was 659, where the negative class amounted to 353 data, while the positive class amounted to 306 data. The data obtained were divided into 2 types of training data and test data with a proportion of 80 % and 20 %, respectively, as can be seen in Table 3. Fig. 3 is an example of the signal data obtained, where the x-axis is the length of time

taken in seconds, while the y-axis is the E-nose output value is in the form of ADC. From the figure, it remains not possible to conclude the difference between positive and negative data. Therefore, statistical parameter features extraction and detection algorithms were developed to help distinguish between positive and negative data.

### 3.3 Results of statistical parameter feature extraction

This section compares the 3 statistical parameter combinations of feature extraction as presented in

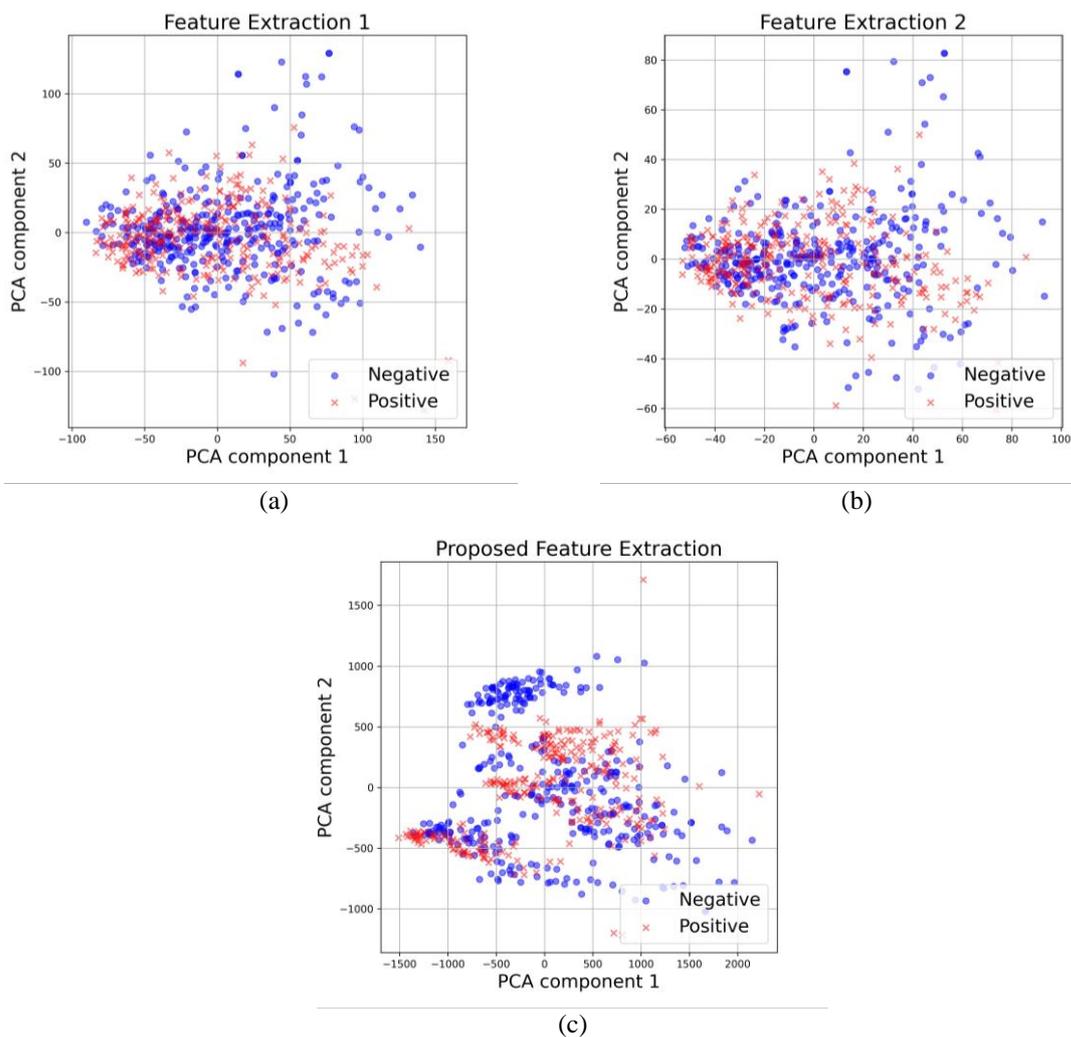


Figure. 4 The comparison of data visualization using different combination of statistical parameters: (a) Feature extraction 1, (b) Feature extraction 2, and (c) Feature extraction 3

Table 4. Comparison of FCDN model performance with different number of hidden layers and several statistical parameter feature extraction

Number of hidden layer in FCDN	Feature Extraction 1			Feature Extraction 2			Feature Extraction 3		
	Acc	Sens	Spec	Acc	Sens	Spec	Acc	Sens	Spec
2 hidden layer	0.750	0.721	0.775	0.818	0.787	0.845	<b>0.940</b>	<b>0.967</b>	0.915
3 hidden layer	0.735	0.689	0.775	0.788	0.787	0.789	0.932	0.934	<b>0.930</b>
4 hidden layer	0.780	0.738	0.817	0.796	0.803	0.789	0.932	0.967	0.901

Table 5. Comparison of confusion matrix with different number of hidden layers and several statistical parameter feature extraction

Number of hidden layer in FCDN	Feature extraction	TP	FN	TN	FP
2 hidden layer	Feature extraction 1	44	17	55	16
	Feature extraction 2	48	13	60	11
	Feature extraction 3	<b>59</b>	<b>2</b>	65	6
3 hidden layer	Feature extraction 1	42	19	55	16
	Feature extraction 2	48	13	56	15
	Feature extraction 3	57	4	<b>66</b>	<b>5</b>
4 hidden layer	Feature extraction 1	45	16	58	13
	Feature extraction 2	49	12	56	15
	Feature extraction 3	<b>59</b>	<b>2</b>	64	7

Table 6. Comparison of FCDN models using indicators value

Number of hidden layer in FCDN	Feature extraction	Accuracy	Number of parameters	Normalized of number of parameters	Indicator
2 hidden layer	Feature extraction 1	0.750	120,417	11.699	63.301
	Feature extraction 2	0.818	137,473	11.831	69.987
	Feature extraction 3	0.940	90,561	11.414	<b>82.526</b>
3 hidden layer	Feature extraction 1	0.735	64,353	11.072	62.412
	Feature extraction 2	0.788	61,633	11.029	67.759
	Feature extraction 3	0.931	131,425	11.786	81.396
4 hidden layer	Feature extraction 1	0.780	109,345	11.602	66.428
	Feature extraction 2	0.796	190,177	12.156	67.340
	Feature extraction 3	0.932	115,969	11.661	81.520

Table 1 using PCA. PCA is a technique that includes unsupervised learning. PCA performs data reduction, so it can be used to help visualize data. Visualized data make it easy to analyze and determine the next steps. Data visualization for 3 statistical parameter combinations using PCA is illustrated in Fig. 4. Qualitative preliminary analysis shows that Fig. 4 (c), namely feature extraction 3 using 29 statistical parameters, is able to separate positive and negative classes and show a significant separation of data clustering. The quantitative comparison of the performance of these feature extractions is described in more detail in section 3.4.

### 3.4 Result of FCDN training phase dan evaluation

After pre-processing, the next stage is the FCDN training phase. The model generated was then tested using the test data and the results can be seen in Table 4, where Acc is accuracy, Sens is sensitivity, and spec is specificity. In the table, it can be seen that FCDN with a different number of hidden layers and feature extraction 3 produce higher and more consistent accuracy, sensitivity, and specificity value than using feature extraction 1 and 2; it proves that using 29 statistical parameters of feature extraction are able to differentiate data among classes better, on the other hand, by using feature extraction 1 and 2 which are

less able to distinguish positive and negative class data will lead to increased numbers of false positives or false negatives. The highest accuracy and sensitivity were obtained by FCDN with 2 hidden layers and feature extraction 3; it shows that this model has the least number of incorrect predictions and has the best ability to identify positive patients. Meanwhile, the highest specificity is generated by the combination of FCDN with 3 hidden layers and feature extraction 3 with a value of 0.930; it shows that this model has the best ability to detect negative

Table 7. Selected FCDN model architecture stored in e-nose

Layer	Number of neurons
Input layer	145
1 <sup>st</sup> hidden layer	224
2 <sup>nd</sup> hidden layer	256
Output layer	2

Table 8. Comparative study of the proposed method

Author	Methodology	Accuracy
[13]	LDA	0.727
[12]	SVM	0.871
[14]	CNN	0.879
<b>Proposed method</b>	<b>FCDN 2 hidden layer + 29 statistical parameter feature extraction</b>	<b>0.940</b>

patients.

A more in-depth evaluation was carried out using a confusion matrix as shown in Table 5. FCDN with various numbers of hidden layers using feature extraction 3 has a smaller FN than the others. A low number of FN is highly significant because incorrectly predicting a positive subject will make the subject not receive the right treatment and also increase the risk of spreading viral respiratory infections. FCDN with various numbers of hidden layers using feature extraction 3 also generates a smaller number of FP than the others; it is also important so that negative subjects do not receive over-treatment.

### 3.5 Results of model selection using indicator

In the previous section, 9 FCDN models were obtained, then one FCDN model was selected and stored in the E-nose to help detect viral respiratory infections. The model was selected automatically using the indicators as presented in Table 6, where the model chosen is FCDN with 2 hidden layers using feature extraction 3. This FCDN model has the best accuracy compared to the others, namely 0.940 and the number of parameters is not too high compared to the others, namely 90,561, so the complexity is not too high. The lowest complexity is obtained by FCDN with 3 hidden layers using feature extraction 2, but the indicator value is low because the accuracy is only 0.788. The complete architecture of the FCDN model stored in the E-nose can be seen in Table 7.

This study also compares the best model of the proposed method with previous studies using the same data in terms of accuracy. The comparison of the results can be seen in Table 8, where the proposed method produces the best performance with an accuracy of 0.940 compared to LDA, SVM, and CNN which produces an accuracy of 0.727, 0.871, and 0.879, respectively.

## 4. Conclusion

This study proposes a new approach for detecting viral respiratory infections using a sample of sweat from the armpit using an E-nose consisting of five metal-oxide sensors. By using this approach, the detection process becomes faster, less expensive, painless, and non-infectious. Analysis using PCA shows that the 29 statistical parameters proposed in the feature extraction stage can better separate the positive and negative class data. The combination of FCDN which consists of 2 hidden layers and 29 statistical parameters generates the best results with accuracy, sensitivity, and specificity of 0.940, 0.967, and 0.915, respectively. This FCDN model is also the

model chosen to be stored in the E-nose because this model contains the largest indicator value based on the calculation of the number of indicators, a trade-off between the complexity and accuracy of the model. For future works, we will evaluate the effect of correlated gases for detecting viral respiratory infections.

## Conflicts of interest

The authors declare no conflict of interest.

## Author contributions

Conceptualization, Malikhah; methodology, Malikhah; software, Malikhah; validation, Malikhah; writing—original draft preparation, Malikhah; writing—editing, Malikhah; assisted in methodology, M. Syauqi Hanif Ardani and Doni Putra Purbawa; writing—review, Shoffi Izza Sabilla, Kelly Rossa Sungkono, Riyanarto Sarno, Chastine Fatichah, Dwi Sunaryono, Arief Bakhtiar, Libriansyah, Cita RS. Prakoeswa, Damayanti Tinduh, and Yetti Hernaningsih; supervision, Riyanarto Sarno, Sozo Inoue, Chastine Fatichah, Dwi Sunaryono, Arief Bakhtiar, Libriansyah, Cita RS. Prakoeswa, Damayanti Tinduh, and Yetti Hernaningsih; proposes problem ideas, Riyanarto Sarno.

## Acknowledgments

This research was funded by AUN/SEED-Net under Special Program for Research Against COVID-19 (SPRAC), the Indonesian Ministry of Education and Culture under Penelitian Terapan Unggulan Perguruan Tinggi (PTUPT) Program, and Institut Teknologi Sepuluh Nopember (ITS) under project scheme of the Publication Writing and IPR Incentive Program (PPHKI).

## References

- [1] Forum of International Respiratory Societies, *The Global Impact of Respiratory Disease*, Second Edition. European Respiratory Society, Sheffield, 2012.
- [2] H. R. V. Doorn and H. Yu, "Viral Respiratory Infections", *Hunter's Tropical Medicine and Emerging Infectious Diseases*, p. 284, 2020.
- [3] M. Shirasu and K. Touhara, "The Scent of Disease: Volatile Organic Compounds of The Human Body Related to Disease and Disorder", *Journal of Biochemistry*, Vol. 150, No. 3, pp. 257–266, 2011.
- [4] S. Sethi, R. Nanda, and T. Chakraborty, "Clinical Application of Volatile Organic Compound Analysis for Detecting Infectious

- Diseases”, *Clinical Microbiology Reviews*, Vol. 26, No. 3, pp. 462–475, 2013.
- [5] I. Mogilnicka, P. Bogucki, and M. Ufnal, “Microbiota and Malodor—etiology and Management”, *International Journal of Molecular Sciences*, Vol. 21, No. 8, pp. 1–21, 2020.
- [6] Y. Li, Y. Hu, Y. Yu, X. Zhang, B. Li, J. Wu, J. Li, Y. Wu, X. Xia, H. Tang, and J. Xu, “Positive Result of Sars-Cov-2 in Faeces and Sputum from Discharged Patients with COVID-19 in Yiwu, China”, *Journal of Medical Virology*, Vol. 92, No. 10, pp. 1938–1947, 2020.
- [7] D. R. Wijaya, R. Sarno, and E. Zulaika, “DWTLSTM for Electronic Nose Signal Processing in Beef Quality Monitoring”, *Sensors and Actuators, B: Chemical*, Vol. 326, p. 128931, 2021.
- [8] A. D. Wilson and M. Baietto, “Applications and Advances in Electronic-Nose Technologies”, *Sensors*, Vol. 9, No. 7, pp. 5099–5148, 2009.
- [9] A. D. Wilson, “Applications of Electronic-Nose Technologies for Noninvasive Early Detection of Plant, Animal and Human Diseases”, *Chemosensors*, Vol. 6, No. 4, p. 45, 2018.
- [10] K. D. Heer, M. G. M. Kok, N. Fens, E. J. M. Weersink, A. H. Zwinderman, M. P. C. V. D. Schee, C. E. Visser, M. H. J. V. Oers, and P. J. Sterk, “Detection of Airway Colonization by *Aspergillus Fumigatus* by Use of Electronic Nose Technology in Patients with Cystic Fibrosis”, *Journal of Clinical Microbiology*, Vol. 54, No. 3, p. 569, 2016.
- [11] M. P. Brekelmans, N. Fens, P. Brinkman, L. D. Bos, P. J. Sterk, P. P. Tak, and D. M. Gerlag, “Smelling the Diagnosis: The Electronic Nose as Diagnostic Tool in Inflammatory Arthritis. A Case-Reference Study”, *PloS one*, Vol. 11, No. 3, p. e0151715, 2016.
- [12] T. Saidi, O. Zaim, M. Moufid, N. E. Bari, R. Ionescu, and B. Bouchikhi, “Exhaled Breath Analysis using Electronic Nose and Gas Chromatography–mass Spectrometry for Non-invasive Diagnosis of Chronic Kidney Disease, Diabetes Mellitus and Healthy Subjects”, *Sensors and Actuators B: Chemical*, Vol. 257, pp. 178–188, 2018.
- [13] H. Y. Yang, H. Y. Peng, C. J. Chang, and P. C. Chen, “Diagnostic Accuracy of Breath Tests for Pneumoconiosis using an Electronic Nose”, *Journal of Breath Research*, Vol. 12, No. 1, p. 016001, 2018.
- [14] M. Rivai, F. Kurniawan, Z. Muchidin, and D. Aulia, “Identification of Diabetes through Urine Using Gas Sensor and Convolutional Neural Network”, *International Journal of Intelligent Engineering and Systems*, Vol. 15, No. 1, pp. 520–529, 2022.
- [15] B. Kim, J. Kim, H. Chae, D. Yoon, and J. W. Choi, “Deep Neural Network-based Automatic Modulation Classification Technique”, In: *Proc. of 2016 International Conference on Information and Communication Technology Convergence (ICTC)*, Jeju, Korea, pp. 579–582, 2016.
- [16] S. I. Sabilla, R. Sarno, K. Triyana, and K. Hayashi, “Deep Learning in a Sensor Array System based on The Distribution of Volatile Compounds from Meat Cuts using GC–MS Analysis”, *Sensing and Bio-Sensing Research*, Vol. 29, p. 100371, 2020.
- [17] W. Harsono, R. Sarno, and S. I. Sabilla, “Recognition of Original Arabica Civet Coffee based on Odor using Electronic Nose and Machine Learning”, In: *Proc. of 2020 International Seminar on Application for Technology of Information and Communication: IT Challenges for Sustainability, Scalability, and Security in the Age of Digital Disruption, iSemantic*, Semarang, Indonesia, pp. 333–339, 2020.
- [18] S. Wakhid, R. Sarno, S. I. Sabilla, and D. B. Maghfira, “Detection and Classification of Indonesian Civet and Non-Civet Coffee based on Statistical Analysis Comparison Using E-Nose”, *International Journal of Intelligent Engineering and Systems*, Vol. 13, No. 4, pp. 56–65, 2020, doi: 10.22266/ijies2020.0831.06.
- [19] R. Sarno, K. Triyana, S. I. Sabilla, D. R. Wijaya, D. Sunaryono, and C. Fatichah, “Detecting Pork Adulteration in Beef for Halal Authentication Using an Optimized Electronic Nose System”, *IEEE Access*, Vol. 8, pp. 221700–221711, 2020.
- [20] M. Malikhah, R. Sarno, and S. I. Sabilla, “Ensemble Learning for Optimizing Classification of Pork Adulteration in Beef Based on Electronic Nose Dataset”, *International Journal of Intelligent Engineering and Systems*, Vol. 14, No. 4, pp. 44–55, 2021, doi: 10.22266/ijies2021.0831.05.
- [21] A. Altinors, F. Yol, and O. Yaman, “A Sound based Method for Fault Detection with Statistical Feature Extraction in UAV motors”, *Applied Acoustics*, Vol. 183, p. 108325, 2021.
- [22] A. Jadon, A. Varshney, and M. S. Ansari, “Low-Complexity High-Performance Deep Learning Model for Real-Time Low-Cost Embedded Fire Detection Systems”, *Procedia Computer Science*, Vol. 171, pp. 418–426, 2020.
- [23] D. Hu, Y. Zhang, L. He, and J. Wu, “Low-

- Complexity Deep-Learning-Based DOA Estimation for Hybrid Massive MIMO Systems with Uniform Circular Arrays”, *IEEE Wireless Communications Letters*, Vol. 9, No. 1, pp. 83–86, 2020.
- [24] V. N. G. Raju, K. P. Lakshmi, V. M. Jain, A. Kalidindi, and V. Padma, “Study the Influence of Normalization/Transformation Process on the Accuracy of Supervised Classification”, In: *Proc. of the 3rd International Conference on Smart Systems and Inventive Technology, ICSSIT*, Tirunelveli, India, pp. 729–735, 2020.
- [25] L. Li and A. Talwalkar, “Random Search and Reproducibility for Neural Architecture Search”, In: *Proc. of The 35th Uncertainty in Artificial Intelligence Conference*, Tel Aviv, Israel, pp. 367–377, 2020.
- [26] I. H. Witten, E. Frank, M. A. Hall, and C. J. Pal, *Data Mining: Practical Machine Learning Tools and Techniques*, Fourth Edition, Elsevier, Amsterdam, Netherlands, 2016.
- [27] S. Mehta, C. Paunwala, and B. Vaidya, “CNN based Traffic Sign Classification using Adam Optimizer”, In: *Proc. of 2019 International Conference on Intelligent Computing and Control Systems (ICCS)*, Madurai, India, pp. 1293–1298, 2019.
- [28] B. E. Guedes, “On the Natural Logarithm Function and its Applications”, *viXra:1503.0058*, No. 2, pp. 1–7, 2015.
- [29] S. Zolfani, M. Yazdani, D. Pamucar, and P. Zaraté, “A VIKOR and TOPSIS Focused Reanalysis of The MADM Methods based on Logarithmic Normalization”, *arXiv preprint arXiv:2006.08150*, Vol. 18, No. 3, pp. 341–355, 2020.