Chemometrics and Intelligent Laboratory Systems 166 (2017) 69–80

Contents lists available at ScienceDirect

Chemometrics and Intelligent Laboratory Systems

journal homepage: www.elsevier.com/locate/chemolab

Efficient android electronic nose design for recognition and perception of fruit odors using Kernel Extreme Learning Machines

Ayşegül Uçar *, Recep Özalp **

Mechatronics Engineering Department, Firat University, 23119, Elazig, Turkey

ARTICLE INFO

Article history:
Received 12 March 2017
Received in revised form
21 April 2017
Accepted 14 May 2017

Keywords:
Odor recognition
Android electronic nose
Kernel extreme learning machines

ABSTRACT

This study presents a novel android electronic nose construction using Kernel Extreme Learning Machines (KELMs). The construction consists of two parts. In the first part, an android electronic nose with fast and accurate detection and low cost are designed using Metal Oxide Semiconductor (MOS) gas sensors. In the second part, the KELMs are implemented to get the electronic nose to achieve fast and high accuracy recognition. The proposed algorithm is designed to recognize the odor of six fruits. Fruits at two concentration levels are placed to the sample chamber of the electronic nose to ensure the features invariant with the concentration. Odor samples in the form of time series are collected and preprocessed. This is a newly introduced simple feature extraction step that does not use any dimension reduction method. The obtained salient features are imported to the inputs of the KELMs. Additionally, K-Nearest Neighbor (K-NN) classifiers, the Support Vector Machines (SVMs), Least-Squares Support Vector Machines (LSSVMs), and Extreme Learning Machines (ELMs) are used for comparison. According to the comparative results for the proposed experimental setup, the KELMs produced good odor recognition performance in terms of the high test accuracy and fast response. In addition, odor concentration level was visualized on an android platform.

1. Introduction

Humans' olfactory system is complex. When the perceiving membrane in the nose cavity is confronted with different composite compounds, the odor detecting cells in the membrane generate a characteristic odor trace. Signals received by the cells are transmitted to the brain to recognize the odors [1]. The functionality of the smell organ and the details of its mechanism led scientists to develop an olfaction machine. In the past two decades, many researchers have carried out many studies about electronic noses by using inspiration from the olfactory systems of humans [2]. An electronic nose is a multidisciplinary topic and is an instrument formed from an array of electronic chemical sensors with overlapping sensitivity, mechanical components, and a digital system such as a microprocessor, microcontroller, or Field Programmable Gate Array (FPGA) [3,4]. An electronic nose consists of two main parts for the perception and recognition of odor samples. The perception part includes a sensor array. A sensor reacts differently to every odor and produces electrical signals by varying its properties, such as conductance, voltage, and capacity. The recognition part includes pattern recognition methods to extract and process the qualities relating to the signal combination from the sensor array. The characteristic data is the signature relating to each odor stimulus and is saved in a database and is then given as input to a pattern recognition system for detection and recognition. Electronic noses have many applications, including the food and beverage industry [5–7], medical diagnosis [8], space applications [9], agriculture [10], environmental monitoring [11,12], public security [13], psychoanalysis [14], cosmetics sectors [15], etc.

As sensor technology has developed, many types of sensitive sensors such as Surface Acoustic Wave (SAW), optical, Quartz Crystal Microbalance (QCM), Conducting Polymer (CP), and Metal Oxide Semiconductor (MOS) sensors have been exhibited. Many researchers have used the sensor arrays to obtain the characteristic signals relating to gas, vapors, and odors [16,17]. On the other hand, an electronic nose also requires attention to pattern recognition methods to achieve reliable recognition by using signals obtained from sensors [18,19]. The recognition problem is organized as being either a classification or regression problem. In the literature, many electronic nose studies have included pattern recognition methods such as Linear Discriminant Analysis (LDA), Principal Component Analysis (PCA), Quadratic Discriminant Analysis (QDA) [20–23], K-Nearest Neighbor (K-NN) [24], Feed-Forward Neural Networks (FNNs) [25–27], Radial Basis Function Neural Network (RBFNN) [28], Probabilistic Neural Networks

* Corresponding author.
** Corresponding author.
E-mail addresses: agulucar@firat.edu.tr (A. Uçar), recepozalp105@gmail.com (R. Özalp).

http://dx.doi.org/10.1016/j.chemolab.2017.05.013
0169-7439/

(PNNs) [29], Self-Organizing Network (SON) [30], Fuzzy Logic, Fuzzy Neural Networks (FNNs) [31,32], Naïve–Bayes [33], competitive neural networks based on learning vector quantization [34], Support Vector Machines (SVMs) [35,36], and Extreme Learning Machines (ELMs) [37–39].

This paper proposes a new low-cost android electronic nose system containing MOS sensor arrays that is different from systems in the extant literature. The e-nose can be used for quality control, food analysis, and the detection of volatiles released by different compounds, which can be used in many areas of industry. The android nose in this paper is specifically designed to recognize six fruit odors, namely apple, lemon, peach, banana, garlic, and onion. The signals obtained from the MOS sensors have the ability to perceive several fruit odors through processing using an ARM Cortex-M4 STM32F407 microcontroller. In addition, this paper also proposes new and simple feature extraction steps without using a dimension reduction method similar to PCA. Moreover, Kernel Extreme Learning Machines (KELMs) are proposed to obtain fast and reliable results from the developed android nose [40]. The application of the KELMs method for electronic nose applications in this paper is the first of its kind. The performance of K-NN, FNNs, SVMs, Least Squares Support Vector Machines (LSSVMs), and ELMs are used as comparisons for the proposed electronic nose [41–45]. Moreover, odor concentration levels are visualized on an android platform remotely using Bluetooth.

The FNNs are capable of approximating a nonlinear function by nonlinear mappings using input samples. The randomly generated initial weights and bias parameters of FNNs are iteratively updated by gradient-based learning algorithms. Hence, the algorithm commonly gets trapped in local minima. Moreover, the FNNs have a very slow learning speed and need a number of iterative learning steps in order to obtain better learning accuracy. SVMs propose a convex optimization problem performing both structural risk minimization and empirical risk minimization simultaneously to obtain a unique optimum solution. Among the used pattern recognition methods, ELMs are the most preferred classifier because of their fast learning, high performance, real-time implementation, and low human intervention [44,45]. Conventional ELMs have the form of single layer FNNs. In ELMs, the weights of the hidden nodes and biases are randomly chosen and the output weights are analytically determined. Thanks to the least square loss, ELMs construct a hidden matrix with a nonlinear differentiable activation function and analytically calculate the weights between the output layer and the hidden layer. In this study, an improved ELM version, KELM, is proposed to classify more quickly and with higher accuracy than both the ELM and the other pattern recognition methods. KELM uses a kernel matrix in hidden layers, satisfying the Mercer Condition [40].

The rest of this paper is organized as follows. In Section 2, a review of the used pattern recognition methods is given. Section 3 introduces the proposed electronic nose system, system operating stages, the preprocessing and feature extraction steps, and the visualization process of odor concentration. Results and discussion relating to the comparative experimental results are described in Section 4. Section 5 concludes the paper.

2. Used pattern recognition methods

This section briefly discusses the principles of FNN, SVM, LSSVM, ELM, and KELM for classification problems.

2.1. Support vector machines

Consider a given set of N samples of training data $\{x^i, y^i\}_1^N$, $x \in \Re^n$, $y \in \{-1,1\}$.

SVM builds the best separating hyperplanes $y^i[w^T g(x^i) + b] \geq 1$ in a high dimensional space [42]. The inputs are mapped to a feature space by a linear/nonlinear mapping function $g(.): x \rightarrow g(x^i)$. The hyperplanes are represented by the normal vector of the separating hyperplane $w \in \Re^n$ and its offset $b \in \Re$. In order to determine the best separating hyperplane, it is minimized to the term of w that defines the distance of the points closest to the hyperplanes of two classes so that the margin $\frac{2}{w}$ is maximized [46]. In addition, SVM allows for misclassified training samples in the margin, thereby satisfying $y^i[w^T g(x^i) + b] \geq 1 - \xi^i$, where ξ^i is the absolute training error. In order to minimize training errors and maximize the margin, the primal form of SVM is formulated as:

$$\min_{w,b,\xi} J(w, b, \xi) = C \sum_{i=1}^{N} \xi^i + \frac{1}{2} \| w \|^2 \tag{1}$$

$$\text{s.t. } \forall\, i\text{: } y^i \left[w^T g(x^i) + b \right] \geq 1 - \xi^i,\ \xi^i \geq 0 \tag{1a}$$

where the first term and second term represent empirical risk and the structural risk, respectively, and C is a user defined regularization parameter achieving tradeoff between the margin and the training error.

Applying the Lagrange multiplier method [42] to the primal problem in (1), its dual formulation is obtained as:

$$\max_{\lambda} J(\lambda) = -\frac{1}{2} \sum_{i,j=1}^{N} \lambda^i \lambda^j y^i y^j K\left(x^i, x^j\right) + \sum_{i=1}^{N} \lambda^i \tag{2}$$

$$\text{s.t. } \forall\, i\text{: } \sum_{i=1}^{N} y^i \lambda^i = 0,\ 0 \leq \lambda^i \leq C \tag{2a}$$

where $\lambda^i \geq 0$ are Lagrange multipliers and $K(x^i, x^j) = g(x^i) g(x^j)^T$ is defined as a kernel function. $K(x^i, x^j)$ is a symmetric and positive definite function satisfying the Mercer condition. Linear and Gaussian kernels are given, respectively, as:

$$K\left(x^i, x^j\right) = g\left(x^i\right) g\left(x^j\right)^T = \left(x^i\right)\left(x^j\right)^T \tag{3}$$

$$K(x^i, x^j) = exp(-\| x^i - x^i \|^2 / 2\sigma^2) \tag{3a}$$

where σ is the spread parameter of the kernel. When the quadratic programming problem in (2) is solved, the best separating surface of SVM is computed as:

$$l(x) = \sum_{support\ \ vectors} y^i \lambda^i K\left(x^i, x^j\right) + b \tag{4}$$

where the support vectors refer to the data points x^i of $\lambda^i > 0$.

2.2. Least squares support vector machines

LSSVM builds an altered version of SVM by introducing the square error and equality constraint [43]. The constructed optimization problem is directly solved by a set of linear equations without using quadratic programming. The optimization problem of LSSVM is written as follows:

$$\min_{w,b,\xi} J(w, b, \xi) = C \frac{1}{2} \sum_{i=1}^{N} \left(\xi^i\right)^2 + \frac{1}{2} \| w \|^2 \tag{5}$$

$$\text{s.t. } \forall\, i\text{: } y^i [w^T g\left(x^i\right) + b] = 1 - \xi^i,\ \xi^i \geq 0. \tag{5a}$$

By introducing a Lagrangian function to primal problem in (5a), the solution is equal to solving the linear equation system below:

$$\begin{bmatrix} 0 & -\boldsymbol{Y}^T \\ \boldsymbol{Y} & \boldsymbol{\Omega} + \boldsymbol{C}^{-1}\boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{b} \\ \lambda \end{bmatrix} = \begin{bmatrix} 0 \\ \vec{1} \end{bmatrix} \tag{6}$$

where $\Omega = y^i y^j g(x^i)^T g(x^j) = y^i y^j K(x^i, x^j)$. The separating surface of LSSVM is calculated as in (4).

2.3. Extreme learning machines

ELMs are a specific type of FNNs. The theory of basic ELMs was originally developed for single layer FNNs with H hidden neurons to achieve extremely fast training, fast testing, and better generalization capability in FNNs at both regression and classification problems by means of only matrix calculations without using back propagation methods [44,45]. Later, ELMs were generalized for hierarchal structures.

Given N samples of training data with n dimension $\{x^i, y^i\}_1^N$, $x \in \Re^n$, $y \in \{-1,1\}$, the ELMs are initialized by random weight w_j and bias b_j according to continuous probability distributions. A linear system is generated to determine the output parameters v_j of the conventional ELM. The mathematical model of ELMs with H hidden neurons is as follows:

$$l(x) = \sum_j^H v_j g\left(w_j x^i + b_j\right) = g(x)V,\ i \in \{1,2,\ldots,N\}. \tag{7}$$

where $g(.)$ is a nonlinear piecewise continuous function achieving the universal approximation capability theorems of ELMs [44,45].

ELM constructs one against all forms of m-class classification problems as $Y = [y^1, y^2, \ldots, y^L] \in \Re^{mxN}$. The matrix form of the ELM output is expressed as follows:

$$\hat{Y} = GV \tag{8}$$

$$V = [v_1, v_2, \ldots, v_m]^T_{Hxm} \tag{9}$$

$$G = \begin{bmatrix} g(w_1 x^1 + b_1) & \ldots & g(w_H x^1 + b_H) \\ \vdots & \ldots & \vdots \\ g(w_H x^N + b_1) & \ldots & g(w_H x^N + b_H) \end{bmatrix}_{NxH} \tag{10}$$

where G is the hidden layer output matrix and V is the output weight matrix [44,45].

Regularized ELM is formulated to minimize both training errors and the norm of output weights $\|V\|^2$ as follows:

$$\min_V J(V) = \frac{1}{2}\|V\|^2 + C\frac{1}{2}\sum_{i=1}^{N}\|\xi^i\|^2 \tag{11}$$

$$\text{s.t. } \forall\, i:\ g(x^i)V = y^i - \xi^i \tag{11a}$$

where the definition of C and ξ^i are the same as those of SVM.

The V solution of the dual form of (11) is calculated as [39]:

$$V = G^{\dagger}Y = \begin{cases} \left(G^T G + \frac{I_{HxH}}{C}\right)^{-1} G^T Y & if\ \ N \geq H \\ G^T\left(G^T G + \frac{I_{NxN}}{C}\right)^{-1} Y & if\ \ N < H \end{cases} \tag{12}$$

where $G^{\dagger}$ is the Moore-Penrose generalized inverse of G and I is the identity matrix. In that case, the output function is computed as:

$$l(x) = g(x)V \tag{13}$$

The predicted output labels of ELM are determined as:

$$label(x) = arg_{j=1,\ldots,m}\max l_j(x) \tag{14}$$

where $l_j(x)$ is the output function of the j-th output node, $l(x) = [l_1(x), \ldots, l_m(x)]^T$.

2.4. Kernel extreme learning machines

KELMs apply to the Mercer condition similar to SVMs in cases where $g(x)$ is an unknown feature mapping [40]. Given input vectors x^i and x^j, a kernel function $K(x^i, x^j)$ is used of instead of the dot product $g(x^i)\cdot g(x^j)$.

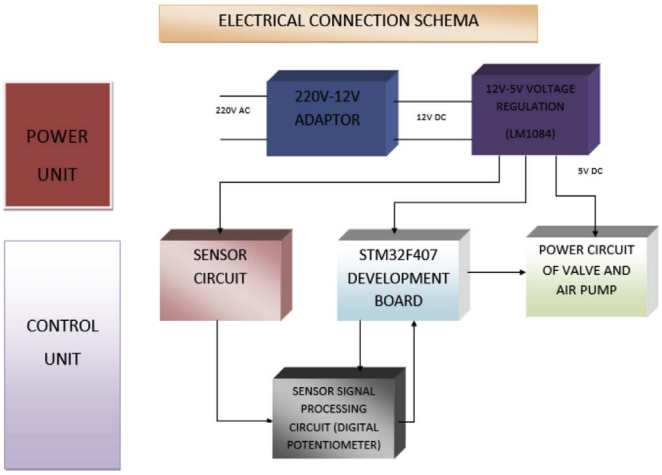


Fig. 1. Block diagram of electronic nose setup.

Fig. 2. The electronic nose system built at Firat University.

Table 1
Characteristics of the five sensors.

Sensor Type	Detection Objects
TGS2600	Carbon monoxide, ethanol, methane
TGS2602	Hydrogen sulfate, ammoniac toluene, ethanol
TGS2610	Propane, methane, hydrogen
TGS2620	Alcohol, organic solvent vapors
TGS826	Ammoniac, ethanol

By using V in (12), the output $l(x)$ of KELM is determined as:

$$l(x) = g(x)\,\mathrm{V} = g(x)G^T\left(G^TG + \frac{I_{NxN}}{C}\right)^{-1}Y = \begin{bmatrix} K(x, x^i) \\ \vdots \\ K(x, x^N) \end{bmatrix}\left(\Omega + \frac{I_{NxN}}{C}\right)^{-1}Y \tag{15}$$

where

$$\Omega = G^TG = \begin{bmatrix} K(x^1, x^1) & \cdots & K(x^1, x^N) \\ \vdots & \ddots & \vdots \\ K(x^N, x^1) & \cdots & K(x^N, x^N) \end{bmatrix}. \tag{16}$$

In KELM, the kernel function (i.e., Gaussian function) is explicitly given. Neither the feature mapping $g(x)$ nor the dimension of feature space H is previously known.

3. Proposed e-nose system and experiments

3.1. Proposed Firat Electronic Nose (FEN)

In this paper, an android electronic nose system was built as the result of rich studies about electronic nose systems constructed at the University of Firat. The block diagram of the developed Firat Electronic Nose (FEN) system and its experimental setup are shown in Figs. 1 and 2, respectively.

The developed FEN system is built in four parts, namely detection, electronics, mechanics, and control box. The detection part includes 5 MOS sensor arrays (TGS2600, TGS2602, TGS2610, TGS2620, TGS826), a type of Taguchi Gas Sensor (TGS), LM35 temperature sensor, and an HIH-4030 humidity sensor. Table 1 shows the detection features of the five gas sensors. The electronics part covers the STM32F407 development kit, MCP42XXX digital potentiometer, and corrector circuit for generating required different voltage values for sensors and valves. In the mechanical part, there is an oxygen tube achieving both air and odors flowing to sensors and solenoid valves. The control box is designed to include all the hardware for the detection and electronics parts and a gas chamber made of plexiglass to replace odor samples.

The FEN system included the design of four Printed Circuit Boards (PCBs). They are the sensor circuit, power circuit, control circuit, and STM32F4 connection circuit. The integrated PCB board sensor circuits are shown in Fig. 3. All circuit design was carried out using the Altium designer software. TGS sensors were selected due to their wide detection range and high sensitivity for

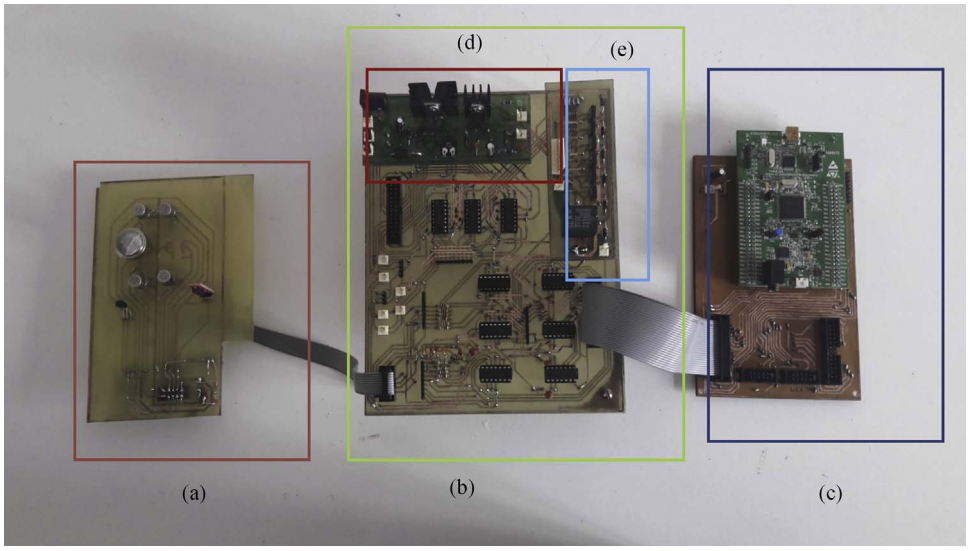


Fig. 3. Designed PCBs (a) sensor circuit, (b) main circuit and sensor data arrangement circuit, (c) the control circuit of STM32F407 discovery kit, (d) 12 V–5 V converter power circuit, (e) valve switching circuit.

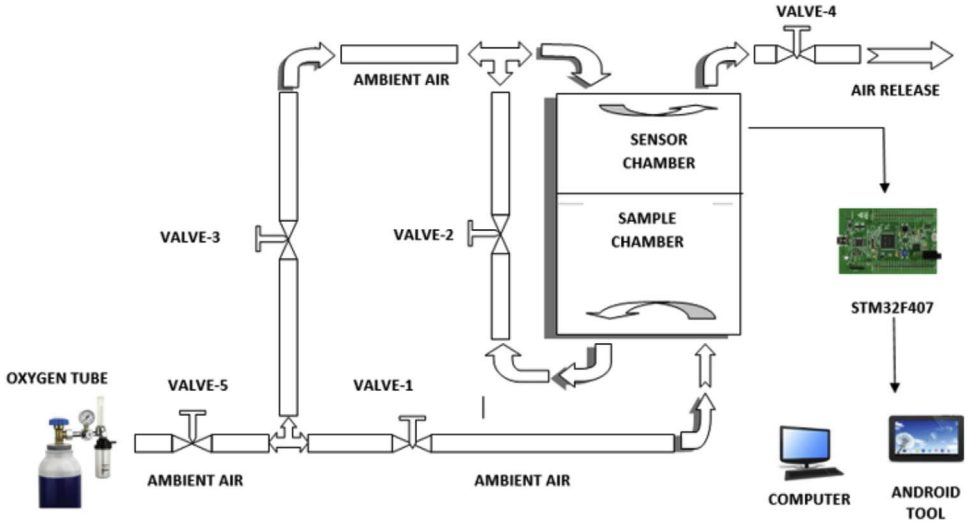


Fig. 4. Electronic nose schema.

monitoring multiple gases and low-cost working voltage of 5 V. However, the software didn't include a special library relating to TGS sensors with six and four legs. Hence, first, a schematic and PCB libraries with respect to the sensor cover and pin locations in the sensor circuit were generated, and, secondly, PCBs were built with a double surface and were isolated. In the circuit, the capacitances were used to reduce the noise effect at the power input. PCB circuits were connected to each other by an Insulation-Displacement Contact (IDC) cable. The system was supplied by 12 V. Those 12 V were directly used for the supply valves. In addition, the voltage was reduced to 5 V for the supply sensors, controller, and the other components using a regularization circuit included an LM1084 component, as can be seen in Fig. 3d. Moreover, cooling and a fan were added to the FEN system.

Since TGS sensors are sensitive to temperature and humidity, the humidity and temperature sensors should be used at the system and the humidity and temperature values of the system should be controlled. Hence, we included an LM35 temperature sensor and HIH-4030 humidity sensor in the FEN system.

The FEN system uses a STM32F407 microcontroller. The evaluation board consists of a 32-bit ARM Cortex-M4 processor core with a processing speed of 72 MHz. The microcontroller was programmed using the Keil IDE. In the STM32F407 microcontroller, the units of Analog Digital Converter (ADC), Universal Synchronous Asynchronous serial Receiver and Transmitter (USART), Serial Peripheral Interface (SPI), DMA (Direct Memory Access), GPIO, and Timer were employed. DMA was used to save the data from environmental units to memory, memory to memory, or from memory to environmental units without using the CPU to get rid of the vanishing data problem in the Multi-Channel ADC mode. Seven of 16 ADC channels on the STM32F407 were used for five gas sensors, LM35 temperature sensor, and the HIH-4030 humidity sensor. The digital potentiometer MCP42XXX was used instead of R_L resistance to determine the value of R_S resistance of the gas sensors using the SPI communication protocol. USART was used to send data from the STM32F407 microcontroller to the computer and android tool by the HCo5 Bluetooth module.

3.2. System operation steps

In the proposed FEN system, the measurements were obtained for approximate fixed heat and humidity values. The system was tested with six fruit odors consisting of apple, lemon, peach, banana, garlic, and onion. The fruits were placed into the sample chamber and odor samples were received. In Fig. 4, the developed electronic nose schema is presented. The system was operated with the following steps:

i. Valve 5, valve 3, and valve 4 were first opened and clean air was applied for 100 s. In this way, undesired gases were purged and the sensor resistance values were compelled to reach to their continuous states.
ii. During the last 35 s, valve 3 and valve 4 were closed and valve 1 and valve 2 were opened at the same time. This allowed the clean air obtained by the oxygen tube to be forwarded to the sample chamber and the odor molecules were moved in the sensor chamber by means of the clean air.
iii. After the convection of odor molecules for 35 s, all valves were closed to lock in the odor molecules in the sensor chamber for 35 s.
iv. At the final stage, valve 5, valve 3, and valve 4 were opened so that sensors were re-cleaned by removing odor molecules and the values of sensor resistances reached a continuous state for 130 s.

3.3. Experimental data, feature extraction, and preprocessing

Each sensor reacts differently to volatiles released by the fruits. Hence, six sensors provide six different outputs in the form of a time series. For odor recognition, the time series should first be further pre-processed to obtain comprehensible signals. Second, some important features from the pre-processed time series should be extracted for odor recognition. Finally, the features should be imported to the KELM inputs.

Fig. 5 shows the block schema of the proposed odor recognition algorithm. In the recognition algorithm, the experimental data consists of time series signals relating to the output voltage

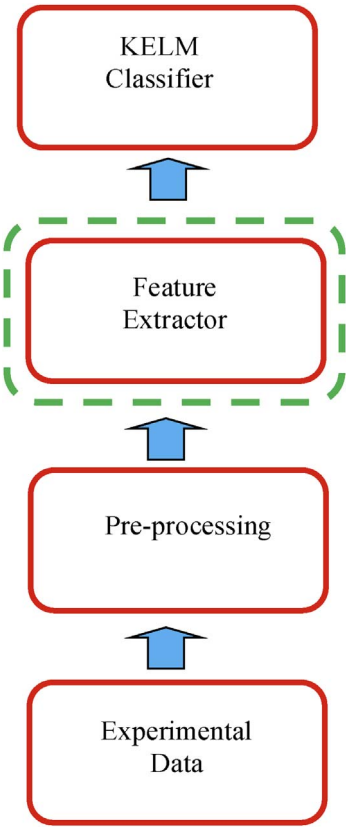


Fig. 5. The block schema of the proposed algorithm.

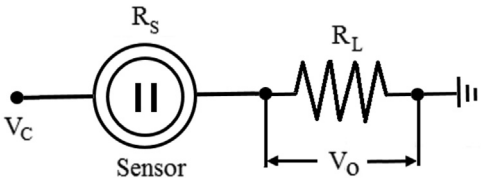


Fig. 6. Basic measurement circuit of gas sensors.

Table 2
Experiment numbers for the training and testing stages.

Odor	Experiment Number	
	Training	Testing
apple	32	16
lemon	32	16
peach	32	16
banana	32	16
garlic	24	10
onion	24	10

responses of all sensors, but it doesn't directly include their concentration levels. However, it is known that if the concentration level changes, the output voltage responses of the sensors change also [24,34]. The output voltage responses of the sensors calculated for different concentration levels are included for the recognition algorithm such that the FEN system is not affected by concentration level and carries out correct recognition of each fruit species. In this paper, the pre-processing and feature extraction steps were generated as:

- Analog data obtained from the output voltage responses of the sensors (e.g., see Fig. 10 in next section) was sampled at 4 s^{-1} frequency during 300 s by means of ADC unit of STM32F407 and a total of 1200 samples were obtained. Fig. 6 gives the basic measurement circuit of the TGS sensors. In Fig. 6, R_L is the load resistor representing an odor and connected in series to the sensor, and R_s is the sensor resistance. The resistance of the sensor changes when exposed to an odor in different concentration levels. A simple voltage dividing rule, sensor response, V_o was calculated as:

$$V_o = \frac{R_L}{R_L + R_s} * V_c \tag{17}$$

where $V_c = 5V$ is the voltage across the sensor element.

- To reduce the measurement noise, the sensor response was filtered as

$$\overline{V}_o(j) = \frac{1}{3}[V_o(j-1) + V_o(j) + V_o(j+1)] \tag{18}$$

- The clean air voltage was calculated for 30 samples before the sample chamber was exactly purged as follows:

$$V_{ca} = \frac{1}{30}\sum_{i=370}^{400} \overline{V}_o(j) \tag{19}$$

- The concentration of volatiles released by fruits make the maximum values of the sensor voltage change. If it is used as raw data for the pattern recognition methods inputs, this causes false odor recognition. To get rid of the varying maximum value problem, the filtered output voltage response of sensors was normalized as follows:

$$V_b(j) = \frac{\overline{V}_o(j) - V_{ca}}{max\overline{V}_o} \tag{20}$$

- 400 samples corresponding to the first 100 s of the output voltage response were used for sensor cleaning. The samples almost did not contain the target odor information duration this time. The remaining data was separated into five regions, including all reactions at both the transient and steady state of sensors. The regions consisted of the samples between 400 and 560, 560–720, 720–880, 880–1,040, and 1040–1200 corresponding to 100–140, 140–180, 180–220, 220–260, and 260–300 in seconds. Each region generated 160 samples at time intervals of 40 s.

- Three important features from each region were extracted. The features are mean, entropy, and variance of the time series data of each region. There are five regions. Each sensor generated 3 features x 5 regions = 15 features for each odor throughout the time interval of 100–300 s.

The inputs and outputs of all mathematical models of the networks in this paper are defined by $\{x^i, y^i\}_1^N$, $x \in \Re^n$, $y \in \{-1,1\}$, respectively. Hence, the inputs can be specifically expressed as:

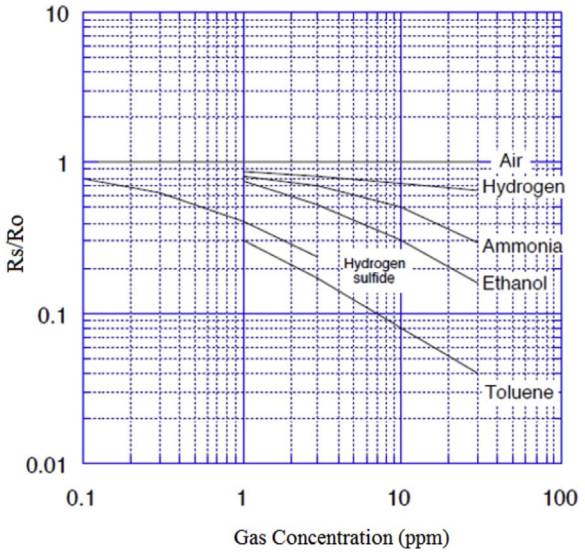
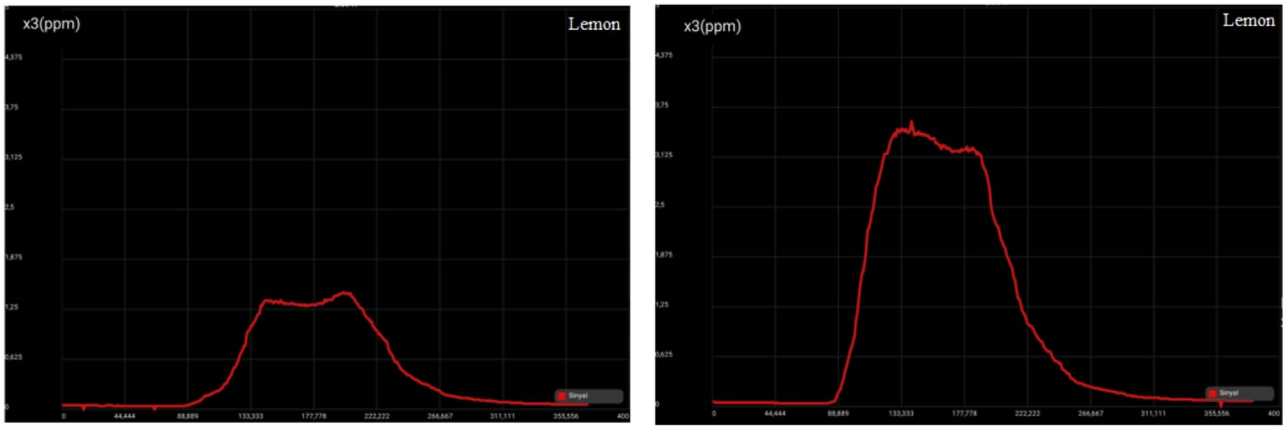


Fig. 7. Sensitivity characteristic of TGS2602 sensor [47].



(a) **(b)**

Fig. 8. Measurements for the concentration level on an android tool for the TGS2602 sensor exposed to (a) half lemon odor and (b) whole lemon odor.

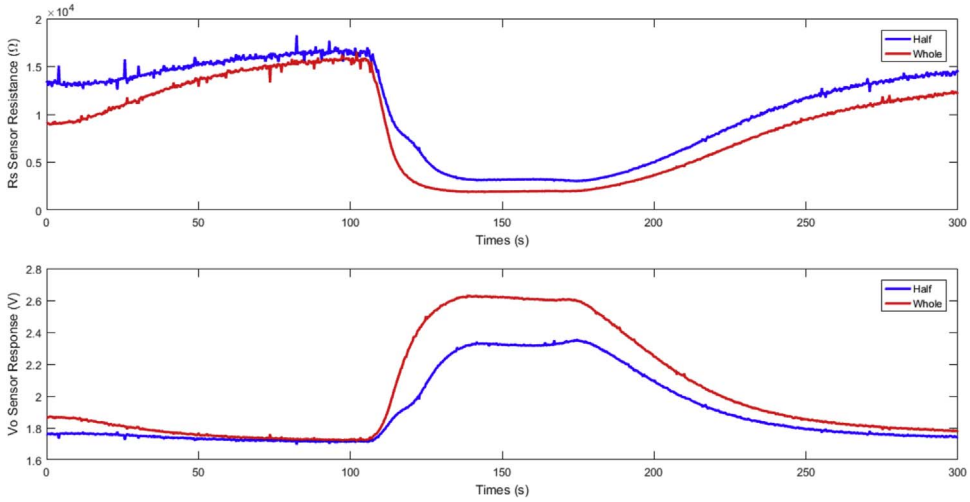


Fig. 9. The sensor resistance variation and output voltage variation for the TGS2602 sensor exposed to (a) half lemon odor and (b) whole lemon odor.

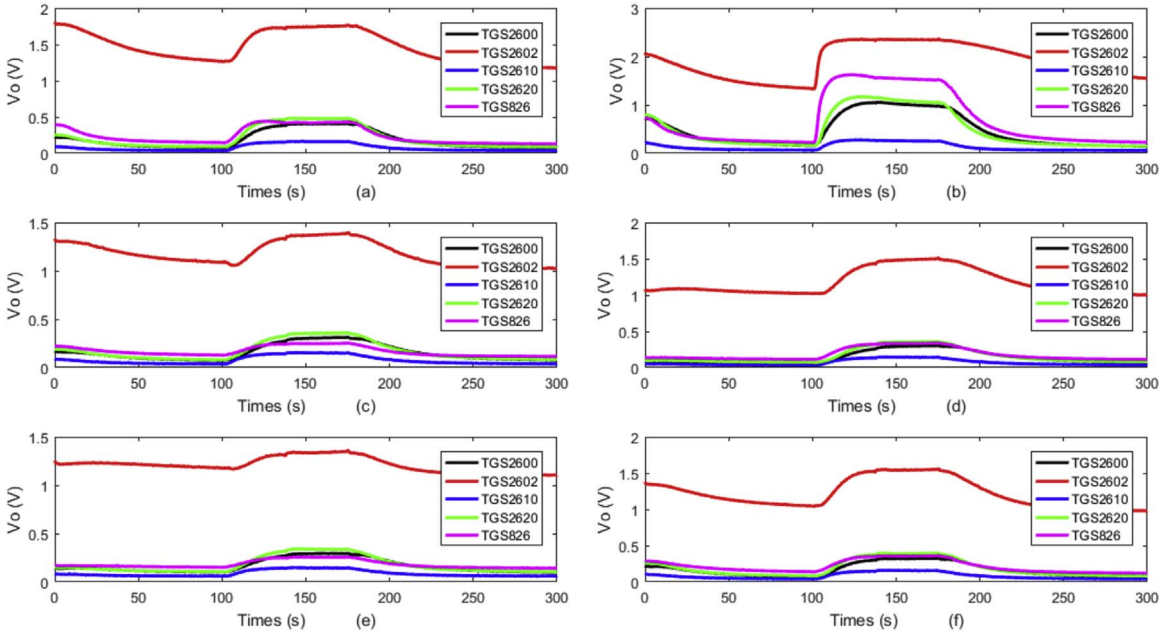


Fig. 10. Experimental measurements for (a) apple, (b) lemon, (c) peach, (d) banana, (e) garlic, and (f) onion.

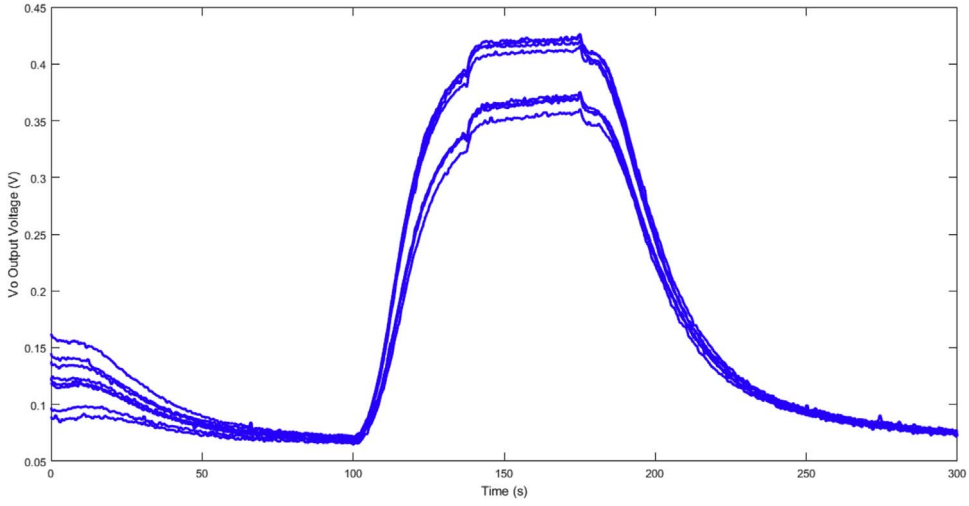


Fig. 11. Repeatable experimental measurements for apple.

$$x^i = \left[x_{TGS2600}{}_j^i \quad x_{TGS2602}{}_j^i \quad x_{TGS2610}{}_j^i \quad x_{TGS2620}{}_j^i \quad x_{TGS826}{}_j^i \right], \quad j: 1, \ldots, 15 \quad \text{and} \quad i: 1, \ldots, N \tag{21}$$

It is clearly seen from (21) that the input number of the networks, n, is 75. In addition, the length of the input vector of the classifiers, N, is equal to the total of the experiment numbers explained in detail in Table 2 in the next section. This number is 152 and 84 for training and testing, respectively.

- The outputs of the networks, y, define only the species of the fruits. In this paper, the networks are capable of classifying odor species, and their outputs don't include concentration information. To achieve the invariance to concentration in the FEN system, the fruits in two different concentration levels were used as half and whole and they were assigned to the same label, i.e., the same species of fruit.

3.4. Visualization of concentration on an android platform

The aim of this paper is not to calculate the concentration levels by using the pattern recognition methods. However, the calculated concentration level using the production information of the sensor and the sensor response obtained in the FEN system are displayed on an android platform for visualization. For this, the production information of all of the sensors is taken into consideration [47]. When exposing to the odor a sensor, both sensor resistance and the concentration level vary. From the sensitivity characteristics representing the variation of the ratio of sensor resistance, R_s, to sensor fresh air resistance, R_o, versus parts-per-million (ppm) gas concentration level, the sensor resistances R_s are calculated first, and then the corresponding concentration level is calculated. In Fig. 7, the sensitivity characteristics of TGS2602 are given as an example. It shows the sensing response of TGS2602 as a function of different ppm concentrations of gases such as ethanol, ammonia, hydrogen sulfide, and toluene.

The fruit odors in Fig. 7 or in product information of all of the used chemical sensors are unseen. However, it can be used as a reference the concentration curve of one of the gases exhibiting a high sensitivity characteristic for a sensor to show the concentration of the other odors. Hence, in this paper, the fruit concentration level is visualized using a reference standard.

4. Results and discussion

In this paper, 34 different experiments were carried out for garlic and onion and 48 different experiments for the other fruits. Table 2 summarizes the experiment numbers. Both training and testing experiments were carried out for two different

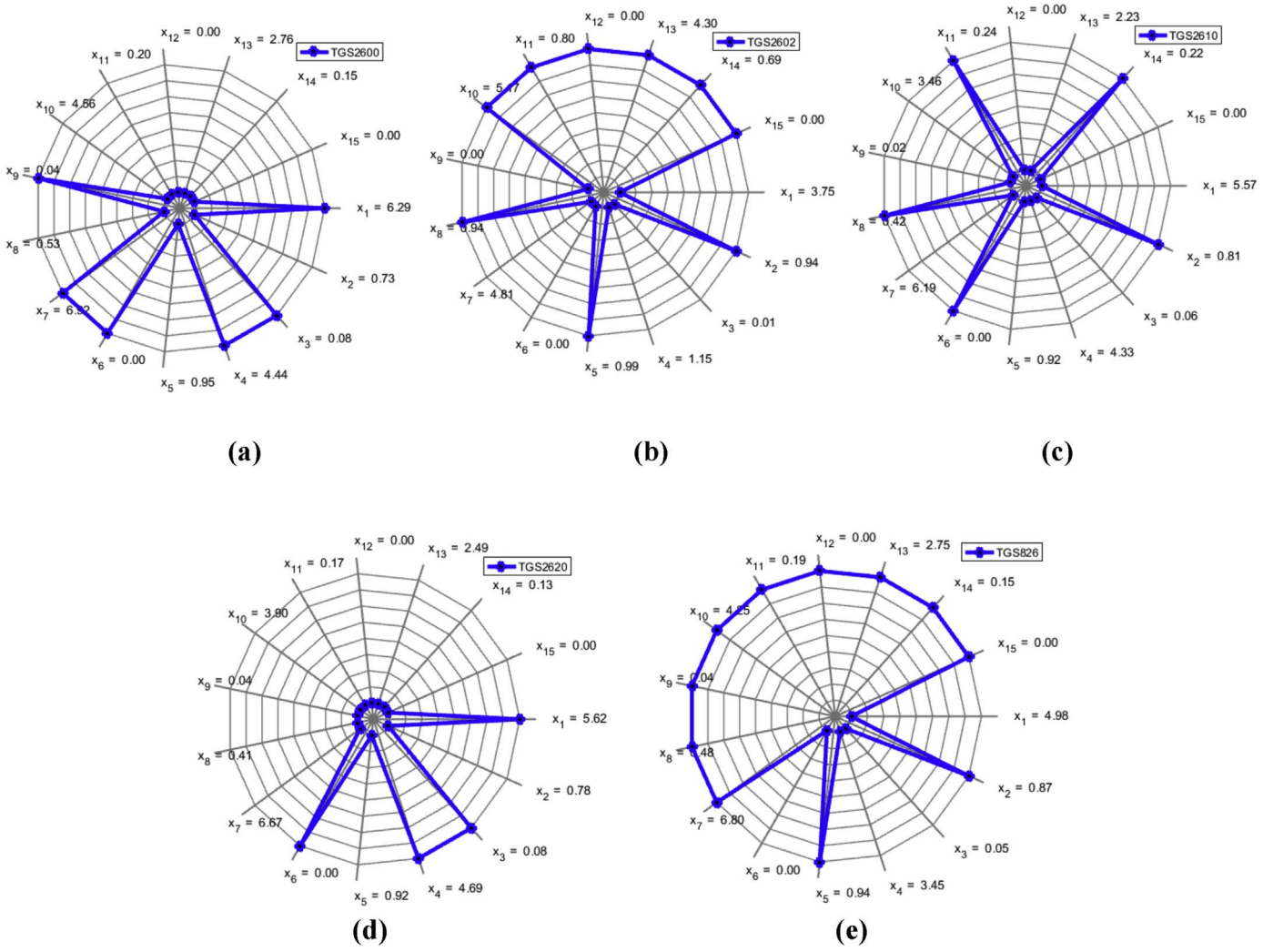


Fig. 12. The radar plots of extracted features from (a) TGS2600, (b) TGS2602, (c) TGS2610, (d) TGS2620, and (e) TGS826 for the lemon odor.

concentrations: odors of half and whole fruits.

In the FEN system, the sensor reactions were obtained in a healthy way. The obtained data from STM32F407 was transferred to a computer and an android tool. The communication with the android tool was constructed using Bluetooth module. In the android tool, a program was written to separately visualize the odor concentration level of all sensors.

In this paper, it was only visualized the concentration information obtained from TGS2602 since the results obtained from the other sensors was similar to that of TGS2602. Fig. 8 gives the sensor response for TGS2602 exposed to a half lemon odor and whole lemon odor on the android tool. The output illustrates the concentration level. In Fig. 8, the concentration level of lemon odor was shown using the ethanol reference standard as cited in the TGS2602 sensor product information sheet. As can be seen from Fig. 8, the odor concentration level of the whole lemon is

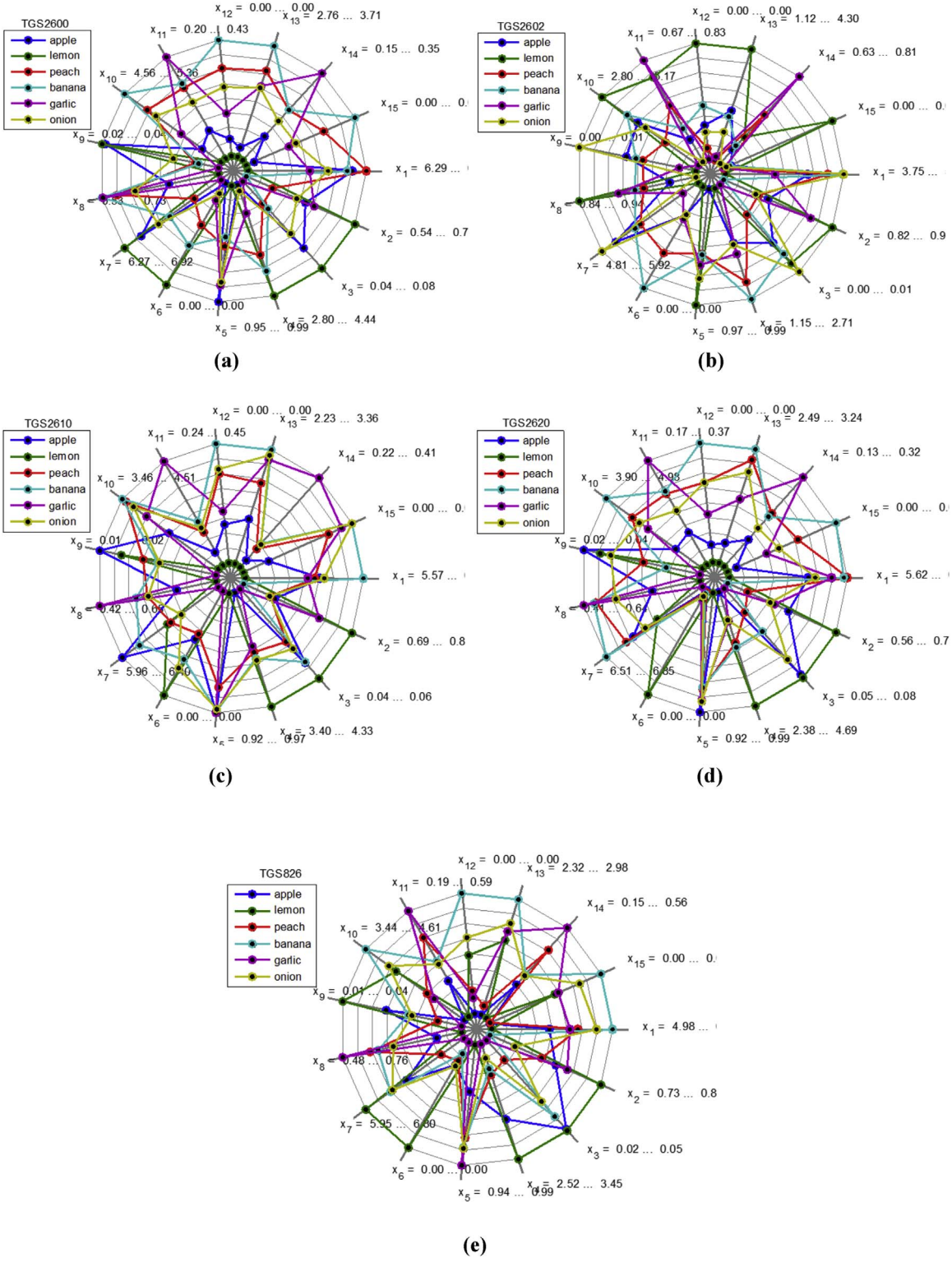


Fig. 13. The radar plots of extracted features from (a) TGS2600, (b) TGS2602, (c) TGS2610, (d) TGS2620, and (e) TGS826 for all fruit odors.

approximately two times of that of the half lemon, which shows that the FEN system operates perfectly.

On the other hand, the sensor resistance variations and its output voltage variations versus time are shown in Fig. 9 for when the TGS2602 sensor was exposed to a half lemon odor and whole lemon odor. It is seen from Fig. 9 that the value of the output voltage increases as the value of the sensor resistance reduces throughout the duration between 100 s and 170 s in which odor molecules are moved in the sensor chamber. In addition, it is observed that the concentration level increases as the sensor resistance reduces.

Fig. 10 shows the characteristic voltage responses of all the sensors against the samples of apple, lemon, peach, banana, garlic, and onion, respectively. Fig. 11 shows reproducible results of the TGS2600 sensor for apple. The results explain that the FEN system is stable and the obtained data from the FEN is correct and repeatable.

Fig. 12(a–e) show the radar plots of the features extracted from the responses of TGS2600, TGS2602, TGS2610, TGS2620, and TGS826, respectively, after exposure to the lemon odor. Radar plots are usually used to observe the differences of selected features (i.e., fingerprints of each sensor) in electronic nose literature. The evaluations were made only on the lemon odor to understand at a glance a clear pattern variation between the different sensors. Fig. 12(a–e) shows 15 feature points represented as $\{x_1, x_1, \ldots, x_{15}\}$ and their values on the radar plot of each sensor. When all features in Fig. 12(a–e) are collected, it is clearly seen that a total of 75 features were obtained from the five sensors. It is verified that the output voltage responses of the sensors used produce distinguishable patterns when exposed to the lemon odor, which makes its recognition easy.

Fig. 13(a–e) separately shows the radar plots of the features extracted from the reactions of TGS2600, TGS2602, TGS2610, TGS2620, and TGS826 exposed to apple, lemon, peach, banana, garlic, and onion, respectively. Each curve with different color in each radar plot of Fig. 13 represents the features obtained from the reactions of only one sensor for six fruits, apple, lemon, peach, banana, garlic, and onion, respectively. There are six different values at each x_i point. The minimum and maximum values are given for each feature on the radar plot. This figure clearly shows that each sensor provides a different response pattern with the six odors and the features obtained from them are separating salient features.

The training and testing stages of K-NN, SVM, LSSVM, ELM, and KELM were run on a computer with a 3.4 GHz Intel i7-2600 K 2 processor in MATLAB. The open source MATLAB interface of LIBSVM was used to implement SVM [48]. The optimal values of the SVM and LSSVM parameters were adjusted by a 10-fold cross validation and the combinations of $1/(2\sigma^2) = [2^4, 2^3, \ldots, 2^{-10}]$ and $C = [2^{12}, 2^{11}, \ldots, 2^{-2}]$. The regularization parameter C of ELM was optimized over $[10^{-10}, 10^{-3}, \ldots, 10^{10}]$, and the number of hidden nodes H was tuned from in the range of $[10, 10^2, 10^3, 10^4]$. In addition, the activation function of the hidden layer of ELM was selected the "radbas" function defined with $G(w, x, b) = exp(-w.\ x + b^2)$ after the sigmoid and sine functions were also tried. Regularization parameter C and kernel parameter σ of KELM were adjusted from the set of $[1, 10^2, 10^4]$ and $[10^{-4}, 10^{-2}, 1, 10^2]$, respectively. The classifiers were compared in terms of the training and testing accuracy and the training and testing speeds.

The training and testing correctness obtained by using K-NN, SVM, LSSVM, ELM, and KELM are given in Table 3. It can be seen from Table 3 that SVM shows high recognition performance with the training correctness value of 100%. For K-NN, LSSVM, ELM, and KELM, good recognition performance with a training accuracy value of 98.8636% was obtained. On the other hand, KELM outperformed the others with a testing accuracy value of 100%

Table 3
Performance comparison of K-NN, SVM, LSSVM, ELM, and KELM as to the training and testing accuracies.

Method	Training Accuracy (%)	Testing Accuracy (%)
K-NN	–	93.75
SVM	100	97.5000
LSSVM	98.8636	97.5000
ELM	98.8636	97.5000
KELM	98.8636	100

Table 4
Performance comparison of K-NN, SVM, LSSVM, ELM, and KELM as to the training and testing times.

Method	Training Time (s)	Testing Time (s)
K-NN	–	0.0163
SVM	0.0102	0.0102
LSSVM	0.0458	0.0240
ELM	0.0025	0.0010
KELM	6.8884e-04	2.8164e-04

Table 5
Confusion matrix relating to the KELM classifier.

Known Class	Predicted Class					
	Apple	Lemon	Peach	Banana	Garlic	Onion
Apple	32	0	0	0	0	0
	16	0	0	0	0	0
Lemon	1	31	0	0	0	0
	0	16	0	0	0	0
Peach	0	1	31	0	0	0
	0	0	16	0	0	0
Banana	0	0	0	32	0	0
	0	0	0	16	0	0
Garlic	0	0	0	0	24	0
	0	0	0	0	10	0
Onion	0	0	0	0	0	24
	0	0	0	0	0	10

It is seen from the results given in Table 4 that the training durations of K-NN, SVM, LSSVM, ELM, and KELM are 0, 0.0102, 0.0458, 0.0025 and 6.8884e-04 s, respectively, while their testing times are 0.0163, 0.0102, 0.0240, 0.0010, and 2.8164e-04 s, respectively. Hence, KELM runs faster than the others. Through these analyses, it is obvious that KELM is an efficient recognition method in comparison to the others. Consequently, KELMs are more appropriate than the others for real time implementations.

All the analyses exhibit that the proposed model is an accurate and fast method for the recognition of fruit odors.

Table 5 gives the confusion matrix KELM in terms of training and testing sample numbers. The diagonal lines give the number of correct classified samples while the others shown mean the number of misclassified samples.

To summarize, it can easily be deduced that the KELM system is strides over and above the K-NN, SVM, LSSVM, and ELM methods in terms of higher testing recognition rates and lower training and testing times for recognizing fruit odors. The extracted features from the proposed features are distinguished as the best for odor recognition.

5. Conclusions

This paper presented an efficient and low cost android electronic nose using KELMs to distinguish it from other previous

works. In the proposed system, low-cost MOS sensors were used to detect and separate the patterns of fruit odors. The main aim of the proposed system is to make use of the amazing features KELM has, such as fast learning speed, better generalization performance, and easier training steps without using time consuming parameters to adjust to recognizing the fruit odors.

In the proposed system, two concentration levels of odor samples were used for obtaining nose invariance. Sensor concentration levels were displayed on an android tool by using a reference standard in the product information sheet of sensor. Time series relating to odor samples were collected and preprocessed. New and simple feature extraction steps were introduced. The obtained comprehensible features were used for the KELM input.

The effectiveness of the proposed system was shown by the sensor reactions, repeatable sensor reactions, radar plots, training and testing accuracy, and the training and testing speeds. Experimental results proved that the proposed system performed significantly well in odor recognition. It was demonstrated that KELM achieved the highest testing accuracy (100%) and the smallest values of training time (6.8884e-04 s) and testing time (2.8164e-04 s).

Meanwhile, a comparative study was carried out on K-NN, SVM, LSSVM, and ELM systems. The experimental results exhibited that KELM surpassed the others in terms of higher recognition performance and shorter training and testing speeds.

In addition, it is notable to express that the proposed system can also be employed for other electronic nose applications outside of the food industry. FPGAs can be used for the electronic nose thanks to fast training and testing properties of KELMs. The next study will be conducted on new feature extraction methods and FPGA.

Acknowledgements

This project was supported by the Turkish Scientific Research Projects Foundation TÜBİTAK. The grant number is 2209.

References

[1] L. Buck, R. Axel, A novel multigene family may encode odorant receptors: a molecular basis for odor recognition, Cell 65 (1991) 175–187.
[2] J.W. Gardner, P.N. Bartlett, A brief history of electronic noses, Sens. Actuators B Chem. 18 (1994) 211–220.
[3] J.W. Gardner, P.N. Bartlett, Electronic Noses: Principles and Applications, Oxford University Press, New York, 1999.
[4] T.C. Pearce, S.S. Schiffman, H.T. Nagle, J.W. Gardner, Handbook of Machine Olfaction: Electronic Nose Technology, John Wiley & Sons, Inc., Hoboken, NJ, USA, 2006.
[5] M.P. Marti, R. Boqúe, O. Busto, J. Guasch, Electronic noses in the quality control of alcoholic beverages, Trac-Trends Anal. Chem. 24 (2005) 57–66.
[6] R.N. Banerjee, B. Tudu, L. Shaw, A. Jana, N. Bhattacharyya, R. Bandyopadhyay, Instrumental testing of tea by combining the responses of electronic nose and tongue, J. Food Eng. 110 (2012) 356–363.
[7] E. Schaller, J.O. Bosset, F. Escher, Electronic noses and their application to food, LWT-Food Sci. Technol. 31 (1998) 305–316.
[8] J.W. Gardner, H.W. Shin, E.L. Hines, An electronic nose system to diagnose illness, Sens. Actuators B Chem. 70 (2000) 19–24.
[9] R. Young, W. Buttner, B. Linnel, R. Ramaesham, Electronics nose for space program application, Sens. Actuator B Chem. 93 (2003) 7–16.
[10] R. Jansen, J.W. Hofstee, H. Bouwmeester, E.V. Henten, Automated signal processing applied to volatile-based inspection of greenhouse crops, Sensors 10 (2010) 7122–7133.
[11] L. Capelli, S. Sironi, R.D. Rosso, Electronic noses for environmental monitoring applications, Sensors 14 (2014) 19979–20007.
[12] R.E. Baby, M. Cabezas, E.N. Walsöe de Reca, Electronic nose: a useful tool for monitoring environmental contamination, Sens. Actuators B Chem. 69 (2000) 214–218.
[13] J. Yinon, Detection of explosive by electronic noses, Anal. Chem. 75 (2003) 98A–105A.
[14] P. Walla, L. Deecke, Odours influence visually induced emotion: behavior and neuroimaging, Sensors 10 (2010) 8185–8197.
[15] B. Dubreuil, M. Bonnefille, S. Neitz, T. Talou, Prospective experiments of e-nose for cosmetics applications: recognition of sweat odors, in: Artificial Chemical Sensing: Olfaction and the Electronic Nose (ISOEN 2001): Proc of the Eighth International Symposium, Pennington, NJ, USA, The Electrochemical Society, New York, NY, USA, 2001.
[16] W. Guoa, F. Gana, H. Kongb, J. Wub, Signal model of electronic noses with metal oxide semiconductor, Chemom. Intell. Lab. Syst. 143 (2015) 130–135.
[17] D. James, S.M. Scott, Z. Ali, W.T. O'Hare, Chemical sensors for electronic nose systems, Microchim. Acta 149 (2005) 1–17.
[18] E.L. Hines, E. Llobet, J.W. Gardner, Electronic noses: a review of signal processing techniques, IEE Proc. Circuit Device Syst. 146 (1999) 297–310.
[19] R. Gutierrez-Osuna, Pattern analysis for machine olfaction: a review, IEEE Sens. J. 2 (2002) 189–202.
[20] J. Brezmes, M.L.L. Fructuoso, E. Llobet, X. Vilanova, I. Recasens, J. Orts, G. Saiz, X. Correig, 2005. Evaluation of an electronic nose to assess fruit ripeness, IEEE Sens. J. 5 (2005) 97–108.
[21] Y. Dai, R. Zhi, L. Zhao, H. Gao, B. Shi, H. Wang, Longjing tea quality classification by fusion of features collected from E-nose, Chemom. Intell. Lab. Syst. 144 (2015) 63–70.
[22] S. Radi, W.S. Ciptohadijoyo, M. Litananda, M.H. Rivai, Purnomo, Electronic nose based on partition column integrated with gas sensor for fruit identification and classification, Comput. Electron. Agric. 121 (2016) 429–435.
[23] X. Peng, L. Zhang, F. Tiana, D. Zhang, A novel sensor feature extraction based on kernel entropy component analysis for discrimination of indoor air contaminants, Sens. Actuators A 234 (2015) 143–149.
[24] S. Kiania, S. Minaeia, M. Ghasemi-Varnamkhastib, A portable electronic nose as an expert system for aroma-based classification of saffron, Chemom. Intell. Lab. Syst. 156 (2016) 148–156.
[25] A.K. Srivastava, Detection of volatile organic compounds (VOCs) using SnO2 gas-sensor array and artificial neural network, Sens. Actuators B Chem. 96 (2003) 24–37.
[26] B. Mumyakmaz, K. Karabaca, An e-nose-based indoor air quality monitoring system: prediction of combustible and toxic gas concentrations, Turk J. Elec. Eng. Comp. Sci. 23 (2015) 729–740.
[27] M.F. Adak, N. Yumusak, Classification of e-nose aroma data of four fruit types by abc-based neural network, Sensors 16 (2016) 1–13.
[28] M. Sriyudthsak, A. Teeramongkolrasasmee, T. Moriizumi, Radial basis neural networks for identification of volatile organic compounds, Sens. Actuators B Chem. 65 (2000) 358–360.
[29] M. Garcia, M. Aleixandre, J. Gutierrez, M.C. Horrillo, Electronic nose for wine discrimination, Sens. Actuators B Chem. 113 (2006) 911–916.
[30] C.D. Natale, A. Macagnano, A. D'Amico, F. Davide, Electronic-nose modeling and data analysis using a self-organizing map, Meas. Sci. Technol. 8 (1997) 1236–1243.
[31] B. Lazzerini, A. Maggiore, F. Marcelloni, FROS: a fuzzy logic-based recogniser of olfactory signals, Pattern Recognit. 34 (2001) 2215–2226.
[32] Z. Haddia, S. Mabroukb, M. Bougrinia, K. Tahria, K. Sghaierb, H. Barhoumib, N. E. Baric, A. Maarefb, N. Jaffrezic-Renaultd, B. Bouchikhia, E-nose and e-Tongue combination for improved recognition of fruit juice samples, Food Chem. 150 (2014) 246–253.
[33] S. Guney, A. Atasoy, Study of fish species discrimination via electronic nose, Comput. Electron. Agric. 119 (2015) 83–91.
[34] S. Omatu, M. Yano, E-nose system by using neural networks, Neurocomputing 172 (2016) 394–398.
[35] X. Honga, J. Wanga, G. Qic, Comparison of semi-supervised and supervised approaches for classification of e-nose datasets: case studies of tomato juices, Chemom. Intell. Lab. Syst. 146 (2015) 457–463.
[36] A. Sanaeifar, S.S. Mohtasebi, M. Ghasemi-Varnamkhasti, H. Ahmadi, Application of MOS based electronic nose for the prediction of banana quality properties, Measurement 82 (2016) 105–114.
[37] S. Qiu, L. Gao, J. Wang, Classification and regression of ELM, LVQ and SVM for E-nose data of strawberry juice, J. Food Eng. 144 (2015) 77–85.
[38] X. Yin, L. Zhang, F. Tian, D. Zhang, Temperature modulated gas sensing e-nose system for low-cost and fast detection, IEEE Sens. J. 16 (2016) 464–474.
[39] L. Zhang, D. Zhang, Evolutionary cost-sensitive extreme learning machine, IEEE Trans. IEEE Trans. Neural Netw. Learn. Syst. 99 (2016) 1–16.
[40] G.B. Huang, H. Zhou, X. Ding, R. Zhang, Extreme learning machine for regression and multiclass classification, IEEE Trans. Syst. Man. Cybern. B Cybern. 42 (2012) 513–529.
[41] S. Haykin, Neural Networks and Learning Machines, third ed., Prentice Hall, Upper Saddle River, USA, 2008.
[42] C. Cortes, V.N. Vapnik, Support vector networks, Mach. Learn 20 (1995) 273–297.
[43] J.A.K. Suykens, J. Vandewalle, Least squares support vector machine classifiers, Neural process. Lett. 9 (1999) 293–300.
[44] G.B. Huang, Q.Y. Zhu, C.K. Siew, Extreme learning machine: a new learning scheme of feedforward neural networks, in: Proc of the 2004 IEEE International Joint Conference on Neural Networks. Budapest, Hungary, 25–29 July 2004, pp. 985–990.
[45] G.B. Huang, Q.Y. Zhu, C.K. Siew, Extreme learning machine: theory and applications, Neurocomputing 70 (2006) 489–501.
[46] A. Uçar, Y. Demir, C. Güzeliş, A penalty function method for designing efficient robust classifiers with input space optimal separating surfaces, Turk. J. Elec. Eng. Comp. Sci. 22 (2014) 1664–1685.
[47] Figaro USA Inc., www.figarosensor.com.
[48] C.C. Chang, C.J. Lin, LIBSVM: a library for support vector machines, ACM Trans. Intell. Syst. Technol. (TIST) 2 (2011) 1–27.