E-Nose Sensor Array Optimization Based on Volatile Compound Concentration Data

Muhammad Asep Subandri and Ricianarto Sarno
Department of Informatics, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia

Abstract. Currently, most e-nose studies are for lab-based applications, the e-nose does not provide access from other places. To be able to implement the internet of things (IoT) technology that is gaining momentum, the e-nose device must be efficient. This study proposes a sensor array optimization technique. If in previous studies using electrical signal data, our study used volatile organic compounds concentration data to minimize the use of sensors. From 10 initial sensors used in the e-nose prototype, only 4 sensors remained. The experimental results showed that by using the KNN algorithm, these 4 sensors were able to predict banana samples with an 80% accuracy rate. When applied to the final e-nose product, the prediction accuracy was 78%.

1. Introduction
E-nose is an electronic instrument created to be able to recognize smell/gas/aroma like a human nose. Even though it is not as sensitive as the human nose, the e-nose is not easily getting fatigued, does not get flu, and can detect odorless harmful gases like carbon monoxide.

Current e-noses are mostly designed for stand-alone applications. To access the sample prediction results, the user must be at the point where the e-nose is located. On the other hand, IoT technology is growing rapidly [1]. Some sectors such as transportation, health, logistics, agriculture, manufacturing, and others have benefited from the application of IoT technology [2]. In general, IoT connects many smart devices, technologies, and applications, then enables the automation of these things [3].

E-nose uses a non-specific chemical sensor array to produce a unique pattern of aroma. The number of sensors on the array can be 8 to 32, or even more. The use of a large number of sensors on the e-nose device makes the computing load high, increases the amount of data that must be sent, and consumes a lot of energy. This is contrary to the characteristics of IoT infrastructure which has resource constraints. In addition, the use of many sensors can also reduce machine learning performance. Sensor array optimization techniques can be done to overcome these problems [4] - [6].

Metal Oxide Semiconductor (MOS) is the most widely used type of chemical sensor in e-nose research. The e-nose device built on previous research [4] - [9], uses the value of the electric voltage (V) or the electrical resistance value (R) produced by the MOS sensor as an input for the pattern recognition system. They do not utilize the potential use of MOS sensors that can detect various volatile organic compounds (1-7 compounds).

In this paper, we try to optimize the e-nose sensor array based on the concentration of volatile organic compounds (VOC) data. As a case study, we built an e-nose to determine the quality of the banana (Musa spp.). The reason behind this is because bananas are the most produced fruit in Indonesia, and there are not many studies that analyze the relationship between changes in the concentration of volatile compounds with the chemical and physical properties of bananas during the maturity process [10].
2. Literature review

Modern e-nose devices generally consist of three main components namely sample room, sensor array, and signal processing system. Signal processing system divided into two subsystems, that is pre-processing signal system and pattern recognition system [11]. Because it mimics the human olfactory system, the components in the e-nose are similar to the components present in the human olfactory system, as well as the process, figure 1 shows these similarities [12], [13].

![Diagram of the human olfactory system and e-nose](image)

The sample chamber serves as a place to put samples, this chamber is made to isolate volatile organic compounds that are generated by samples from the influence of the surrounding environment. The sensor array is a core component of the e-nose device, this component serves to absorb and convert volatile compounds into electrical signals. Gas sensor output data in the form of an electric voltage (V) or electrical resistance (R) then processed by the signal conditioning system to produce only information that is really useful for the pattern recognition system. While the pattern recognition system serves to predict, identify, and classify odor by comparing new odor data with previously recognized odor data.

E-nose does not work by searching for a particular gas molecule or compound, but rather looking for a unique pattern such as "fingerprint" from the analyzed air. This unique pattern is obtained using a chemical sensor array. The sensor array consists of several types of chemical sensors so that when exposed to the aroma each sensor in the array has a special response. For example, odorant A may produce a high response in one sensor and lower responses in others, while on odorant B might produce high readings for sensors other than the one that “took” to odorant A. These different responses are stored in the database, which is then used to train pattern recognition algorithms in odor identification and classification [13].

The e-nose device has attracted enormous attention in the past three decades, with many findings related to e-nose applications originating from various fields of applied science. This discovery provides many benefits for humans [13], such as can be used to detect disease [14], detect explosives [15], detect toxic odorless gases [16], determine the quality of a product [17], identify plant species [18], and so on.

As summarized by [19], the e-nose application is very diverse, ranging from the garden to the battlefield. Since 1993, there have been more than 12,000 e-nose articles published. Most of the applications are in the food industry, such as fruit, meat, milk, wine (drinks), tea, and coffee. It is estimated that there are 5000 publications published since then, almost half of the total e-nose publication [20].

The current e-nose research is laboratory-based, focuses on developing smaller and cheaper devices that can produce better performance. Only a few publications discuss e-nose systems that can monitor samples from other places. Some of the articles found were [9], by adopting WSN technology, the authors installed a wireless communication module on his e-nose device. Using the PCA method to analyze sensor response data (voltage), young, mature, and rotten mango samples can be classified. Then there is [21], using WSN and pattern recognition web software installed on local servers, the authors can predict 12 classes of pollutant samples contained in water with 91% accuracy and 94% using Backpropagation learning algorithm (BP) and Radial-Base algorithms Functions based neural network (RBF), respectively. Then the research article written by [5], the authors proposed the Mobile Electronic Nose system architecture to predict the level of meat decomposition. The proposed system
based on the Internet of Things (IoT) technology, consists of several WSN networks that are integrated with Cloud Computing infrastructure.

3. Methodology

3.1. Making of e-nose prototype

The development began by collecting the tools and materials needed. In this study, the e-nose prototype used an Arduino MEGA 2560 microcontroller and 10 MQ gas sensors placed on the food storage box. A complete list of tools and materials needed can be seen in figure 2.

![E-nose circuit schematic](image)

**Figure 2.** E-nose circuit schematic.

To calculate the gas concentration in the air using the MQ sensor, it takes several steps. Firstly, calculate the sensor resistance value when exposed to gas (Rs), with equations (1) and (2) below:

\[
Rs = \frac{Vc-VRL}{VRL} \times RL 
\]

(1)

\[
VRL = \frac{ADC \times Vc}{1023} 
\]

(2)

Where the ADC is the result of analog to digital conversion obtained from Arduino, Vc is the voltage value on the Arduino measured by a multimeter, VRL is the sensor voltage when exposed to a gas, and RL is the sensor resistance obtained from the datasheet.

After getting the sensor resistance value, the equation (3) can be used to calculate the gas concentration in parts per million (ppm).

\[
C = 10\left(\log\left(\frac{Rs}{Ro}\right)\right)+\alpha
\]

(3)

Ro is the value of sensor resistance in clean air without exposure to any gas, this value can be seen in each MQ gas sensor datasheet. Meanwhile, the values of α, β, and γ were obtained from the curve fitting process using LibreOffice Calc software.
Using the equations and parameters above, Arduino is then programmed to read the gas concentration for 10 minutes with 2 seconds of reading interval. The Arduino is also programmed to clean the sample chamber by turning on the exhaust fan, calibrating the sensor, and sending the readings to the PC / Laptop via a USB cable.

3.2. Sampling
The sample chosen in this study was banana (Musa acuminata × balbisiana). This type of banana produces almost no odor. So, if the e-nose prototype can predict the group of this banana, then it can be ascertained that the prototype can predict any sample. Banana samples are categorized into two groups of ripening stage, unripe and ripe. These two groups are determined by comparing the sample's color and condition. Yellow bananas, bruised, and soft are categorized as mature. While bananas in green, gummy and hard textured are categorized as raw. This collected ground-truth data is 50 samples, consisting of 25 ripe bananas and 25 unripe bananas.

3.3. Sensor array optimization
To get the optimal combination of sensor arrays, the steps taken are as follows:

- The Signal denoising with the Discrete Wavelet Transform (DWT) Daubechies family method. The Daubechies options tested are from db1 to db15.
- Extracting VOC concentration statistical features, such as: maximum value, average, variance, standard deviation, and skewness value. The last four statistical features can be obtained using equations (4), (5), (6), (7).

\[
\text{mean}(\mu) = \frac{1}{N} \sum_{n=1}^{N} x_n \tag{4}
\]
\[
\text{var} = \frac{1}{N-1} \sum_{n=1}^{N} (x_n - \mu)^2 \tag{5}
\]
\[
\text{std}(\sigma) = \sqrt{\text{var}} \tag{6}
\]
\[
\text{skew} = \frac{\frac{1}{N} \sum_{n=1}^{N} (x_n-\mu)^3}{\sigma^3} \tag{7}
\]

Where Xn is the signal value (VOC concentration in ppm) and N is the number of data.

- Feature scaling with the min-max normalization method. This method generates a standard value between 0 and 1, using the following formula (8).

\[
z = \frac{x - \text{min} (x)}{\text{max} (x) - \text{min} (x)} \tag{8}
\]

Where z is the normalized value, while x is the initial value, min is the minimum value, and max is the maximum value of a feature.

- Feature selection using the chi-square statistical method. To calculate the chi-squared statistic, the following equation (9) is used.

\[
X^2 = \sum_{i=1}^{N} \frac{(O_i - E_i)^2}{E_i} \tag{9}
\]

Where X2 is the chi-squared value, Oi is the class i observation frequency, and Ei is the expected class i frequency.

- Performance evaluation with seven classification algorithm approaches, i.e. Logistic Regression, KNN, Decision Tree, Naive Bayes, Support Vector Machine, Linear SVC, and Random Forest. The approach was evaluated using ground-truth data samples that had been taken previously with the k-fold cross-validation method.

- Tuning the parameters of the two best algorithms and tuning the duration of the sample reading.
3.4. Final e-nose product

Based on a combination of sensors selected from the previous sensor array optimization stage, a final e-nose product is made. The final e-nose is configured and programmed like the prototype, except that the final e-nose is installed with a WiFi module to be able to communicate and send data to the server. Then, the tools used such as the Arduino, sensor shield, and the sample chamber box are smaller than the prototype and equipped with a battery.

4. Results

From the ground-truth data acquisition stage using the e-nose prototype, we obtained 54 data on the concentration of volatile compounds (gases) from each sample. From the statistical feature extraction process, 270 features were generated to train the machine learning model.

From the performance evaluation stage, can be seen that: (1) The performance of most machine learning models has increased when using preprocessed data; (2) The performance of the machine learning model varies when using the 25, 20, 15, 10, and 5 features, some of them produce better performance when using 270 features, some worse.

Table 1 shows accuracy comparison between algorithms that use raw data and algorithms that use preprocessed data (Signal preprocessing and Standardization). As we can see, four algorithm performances increased, two decreased, and one did not change.

Table 1. Comparison of raw and preprocessed data accuracy.

| Algorithm | Raw Data Accuracy | SP + SD Data Accuracy |
|-----------|------------------|-----------------------|
| LR        | 0.67             | 0.6                   |
| KNN       | 0.5              | 0.6                   |
| CART      | 0.53             | 0.59                  |
| NB        | 0.57             | 0.44                  |
| SVM       | 0                | 0                     |
| Linear SVC| 0.63             | 0.71                  |
| Random Forest | 0.56             | 0.59                  |

Table 2 shows a comparison of seven algorithms when feature selection performed. The KNN and RF algorithms get the highest performance compared to other algorithms, these two algorithms get the best performance when using 15 features. 15 of these features are produced by 4 MQ sensors, i.e. MQ2, MQ3, MQ6, and MQ7. The details can be seen in Table 3.

Table 2. Algorithm accuracy using selected features.

| Algorithm     | 25 Features Acc. | 20 Features Acc. | 15 Features Acc. | 10 Features Acc. | 5 Features Acc. |
|---------------|------------------|------------------|------------------|------------------|-----------------|
| LR            | 0.62             | 0.62             | 0.61             | 0.61             | 0.54            |
| KNN           | 0.74             | 0.74             | 0.76             | 0.75             | 0.65            |
| CART          | 0.65             | 0.66             | 0.65             | 0.67             | 0.64            |
| NB            | 0.55             | 0.58             | 0.6              | 0.65             | 0.63            |
| SVM           | 0.48             | 0.49             | 0.51             | 0.54             | 0.45            |
| Linear SVC    | 0.64             | 0.63             | 0.62             | 0.62             | 0.63            |
| Random Forest | 0.7              | 0.69             | 0.79             | 0.7              | 0.71            |
Table 3. List of selected features.

| No | 25 Features       | 20 Features     | 15 Features       | 10 Features       | 5 Features       |
|----|-------------------|-----------------|-------------------|-------------------|-----------------|
| 1  | MQ2_LPG_skew      | MQ2_LPG_skew    | MQ2_LPG_skew      | MQ2_LPG_skew      | MQ3_CO_skew     |
| 2  | MQ2_CO_skew       | MQ2_SMOKE_skew  | MQ2_H2_skew       | MQ2_H2_skew       | MQ6_CO_skew     |
| 3  | MQ2_SMOKE_skew    | MQ2_H2_skew     | MQ2_PROPANE_skew   | MQ2_PROPANE_skew   | MQ6_CO_skew     |
| 4  | MQ2_ALCOHOL_skew  | MQ2_PROPANE_skew | MQ3_CO_skew      | MQ3_LPG_skew      | MQ6_LPG_skew    |
| 5  | MQ2_CH4_skew      | MQ3_CH4_var     | MQ3_LPG_skew      | MQ6_CH4_skew      | MQ7_H2_skew     |
| 6  | MQ2_H2_skew       | MQ3_CO_skew     | MQ6_CH4_std       | MQ6_H2_std        | MQ7_H2_skew     |
| 7  | MQ2_PROPANE_skew   | MQ3_LPG_skew    | MQ6_CO_var        | MQ6_H2_std        | MQ6_LPG_skew    |
| 8  | MQ3_CH4_mean      | MQ4_CO_var      | MQ6_H2_std        | MQ6_LPG_std       | MQ7_CO_skew     |
| 9  | MQ3_CH4_var       | MQ6_CH4_std     | MQ6_CH4_var       | MQ7_CO_skew       | MQ7_H2_skew     |
| 10 | MQ3_CO_skew       | MQ6_CH4_skew    | MQ6_LPG_skew      | MQ7_CO_skew       | MQ7_H2_skew     |
| 11 | MQ3_HEXANE_var    | MQ6_H2_std      | MQ7_CO_skew       | MQ7_LPG_var       |                 |
| 12 | MQ3_LPG_skew      | MQ6_H2_skew     | MQ7_H2_skew       |                 |                 |
| 13 | MQ4_CO_var        | MQ6_LPG_std     | MQ7_LPG_mean      |                 |                 |
| 14 | MQ6_CH4_std       | MQ6_LPG_skew    | MQ7_LPG_std       |                 |                 |
| 15 | MQ6_CH4_skew      | MQ7_CO_skew     | MQ7_LPG_var       |                 |                 |
| 16 | MQ6_H2_std        | MQ7_H2_std      |                 |                 |                 |
| 17 | MQ6_H2_skew       | MQ7_H2_skew     |                 |                 |                 |
| 18 | MQ6_LPG_std       | MQ7_LPG_mean    |                 |                 |                 |
| 19 | MQ6_LPG_skew      | MQ7_LPG_std     |                 |                 |                 |
| 20 | MQ7_CO_skew       | MQ7_LPG_var     |                 |                 |                 |
| 21 | MQ7_H2_std        |                 |                 |                 |                 |
| 22 | MQ7_H2_skew       |                 |                 |                 |                 |
| 23 | MQ7_LPG_mean      |                 |                 |                 |                 |
| 24 | MQ7_LPG_std       |                 |                 |                 |                 |
| 25 | MQ7_LPG_var       |                 |                 |                 |                 |

From the tuning parameter stage, it can be seen that the number of neighbors in the KNN algorithm that produces the best performance is 4, 5, 6, 10 (figure 3), while the number of trees in the RF that produces the best performance is 8 (figure 4). Both of these algorithms obtain 76% accuracy.

![Figure 3. KNN parameter tuning result.](image1)

![Figure 4. Random Forest parameter tuning result.](image2)

From the data amount tuning stage, the highest accuracy was obtained on 8 minutes of data if using KNN, and 6 minutes of data if using Random Forest. Table 4 shows the accuracy comparison of these two algorithms.
Table 4. Time tuning results.

|          | 2 Minutes | 4 Minutes | 6 Minutes | 8 Minutes | 10 Minutes |
|----------|-----------|-----------|-----------|-----------|------------|
| KNN      | 0.43      | 0.62      | 0.67      | 0.8       | 0.76       |
| RF       | 0.53      | 0.63      | 0.79      | 0.74      | 0.75       |

The final product of the e-nose device assembled using 4 selected sensors from the sensor array optimization stage can be seen in Figure 5 below.

Figure 5. Final e-nose that is used as a WSN sensor node.

Using new data collected from 50 banana samples, the e-nose pattern recognition system was trained and validated. With the KNN classification method and the 10 fold cross-validation method, the final e-nose is able to predict ripe and unripe bananas with an accuracy rate of 78%.

5. Conclusion

This study proposes an e-nose sensor array optimization technique. The data used as input to the pattern recognition system is the concentration of volatile organic compounds. Using this data, what compounds are produced by a sample and which compound is dominant can be known. From the array sensor optimization stage, the KNN algorithm emerges as an algorithm with the best accuracy, this algorithm gets 80% accuracy when using 15 features (4 MQ sensors) and 8 minutes of data. When applied to final e-nose products, the performance of this algorithm has decreased slightly to 78%.

6. References

[1] L. Reinfurt, U. Breitenbücher, M. Falkenthal, F. Leymann, and A. Riegg, “Internet of things patterns,” Proc. 21st Eur. Conf. Pattern Lang. Programs - Eur. ’16, pp. 1–21, 2016.
[2] P. P. Ray, “A survey on Internet of Things architectures,” J. King Saud Univ. - Comput. Inf. Sci., 2016.
[3] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, “Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications,” IEEE Commun. Surv. Tutorials, vol. 17, no. 4, pp. 2347–2376, 2015.
[4] G. Wei, A. He, and Y. Wang, “A Novel Compressive Sensing Based Electronic Nose of Gas Sensor Dynamic Signals,” pp. 1741–1744, 2012.
[5] D. R. Wijaya, R. Sarno, E. Zulaika, and S. I. Sabila, “Development of mobile electronic nose for beef quality monitoring,” Procedia Comput. Sci., vol. 124, pp. 728–735, 2017.
[6] E. Climent, J. Pelegrí-Sebastia, T. Sogorb, J. Talens, and J. Chilo, “Development of the MOOSY4 eNose IoT for Sulphur-Based VOC Water Pollution Detection,” Sensors, vol. 17, no. 8, p. 1917, 2017.

[7] J. L. Herrero, J. Lozano, J. P. Santos, J. A. Fernandez, and J. I. S. Marcelo, “A Web-Based Approach for Classifying Environmental Pollutants Using Portable E-nose Devices,” IEEE Intell. Syst., vol. 31, no. 3, pp. 108–112, 2016.

[8] A Sanaeifar, “Development and Application of a New Low Cost Electronic Nose for the Ripeness Monitoring of Banana using,” Czech J. Food Sci., vol. 32, no. 6, pp. 538–548, 2014.

[9] L. P. Deshmukh, M. S. Kasbe, T. H. Mujawar, S. S. Mule, and A. D. Shaligram, “A wireless electronic nose (WEN) for the detection and classification of fruits: A case study,” 2016 Int. Symp. Electron. Smart Devices, pp. 174–178, 2016.

[10] A. Sanaeifar, S. S. Mohtasebi, M. Ghasemi-Varnamkhasti, and H. Ahmadi, “Application of MOS based electronic nose for the prediction of banana quality properties,” Measurement, vol. 82, pp. 105–114, Mar. 2016.

[11] Z. Zheng and X. Lin, “Study on Application of Medical Diagnosis by Electronic Nose,” World Sci. Technol., vol. 14, no. 6, pp. 2115–2119, 2012.

[12] A. P. D. Heredia, F. R. Cruz, J. R. Balbin, and W.-Y. Chung, “Olfactory classification using electronic nose system via artificial neural network,” 2016 IEEE Reg. 10 Conf., pp. 3569–3574, 2016.

[13] H. K. Patel, R. H. Austin, J. Barber, and H. K. Patel, The Electronic Nose: Artificial Olfaction Technology. 2014.

[14] Haryanto, R. Sarno, and D. R. Wijaya, “Detection of diabetes from gas analysis of human breath using e-Nose,” in 2017 11th International Conference on Information & Communication Technology and System (ICTS), 2017, vol. 0, pp. 241–246.

[15] P. López, R. Triviño, D. Calderón, A. Arcentales, and A. V. Guzmán, “Electronic nose prototype for explosive detection,” 2017 Chil. Conf. Electr. Electron. Eng. Inf. Commun. Technol. CHILECON 2017 - Proc., vol. 2017–Janua, pp. 1–4, 2017.

[16] V. Ravindra and S. M. Rajbhoj, “Identification of Toxic Gases using Electronic Nose,” 2017 Int. Conf. Comput. Commun. Control Autom., pp. 1–5, 2017.

[17] R. B. Somasagar and A. Kusagur, “Flavor Determination for Milk Quality Assessment using Embedded Electronic Noses,” in 2017 2nd International Conference On Emerging Computation and Information Technologies (ICECIT), 2017, pp. 1–4.

[18] H. Yu, M. Meng, and Y. Yin, “The research on the application of electronic nose in discriminate the rice varieties,” Int. Conf. Adv. Mechatron. Syst. ICAMEchS, pp. 449–454, 2013.

[19] A. D. Wilson and M. Baietto, “Advances in electronic-nose technologies developed for biomedical applications,” Sensors, vol. 11, no. 1, pp. 1105–1176, 2011.

[20] A. Loutfi, S. Coradeschi, G. K. Mani, P. Shankar, and J. B. B. Rayappan, “Electronic noses for food quality: A review,” J. Food Eng., vol. 144, pp. 103–111, 2015.

[21] J. L. Herrero, J. Lozano, J. P. Santos, and J. I. Suárez, “On-line classification of pollutants in water using wireless portable electronic noses,” Chemosphere, vol. 152

Acknowledgments
Authors would like to thank to Institut Teknologi Sepuluh Nopember and the Indonesian Ministry of Research, Technology and Higher Education for supporting the research.