

Received 20 May 2022, accepted 17 June 2022, date of publication 23 June 2022, date of current version 30 June 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3185613

# Classifying Gas Data Measured Under Multiple Conditions Using Deep Learning

HOJUNG LEE<sup>1</sup>, JAEHUI HWANG<sup>1</sup>, HWIN DOL PARK<sup>2</sup>, JAE HUN CHOI<sup>2</sup>,  
AND JONG-SEOK LEE<sup>1</sup>, (Senior Member, IEEE)

<sup>1</sup>School of Integrated Technology, Yonsei University, Incheon 21983, South Korea

<sup>2</sup>Bio-Medical IT Convergence Research Division, Electronics and Telecommunications Research Institute, Daejeon 34129, South Korea

Corresponding author: Jong-Seok Lee (jong-seok.lee@yonsei.ac.kr)

This work was supported by the Ministry of Science and Information and Communications Technologies (ICT) (MSIT), South Korea, Intelligent E-Nose Development for Gas Detection to Sub-ppb Level, under Grant 2020-0-01106.

**ABSTRACT** Gas classification is a machine learning problem that is important for various applications including monitoring systems, health care, public security, etc. Since measuring the characteristic of gas molecules is greatly affected by external factors such as wind speed and the internal setting of detecting sensors, classification should be done by taking into account the combination of these individual factors, which we call a *condition* in this paper. In particular, when classifying gas data measured under multiple conditions, the data from each condition need to be integrated, which we call *multi-conditioned gas classification*. While there have been some studies on gas classification for a single condition, no previous approach deals with the multi-conditioned gas classification problem to the best of our knowledge. In this paper, we propose a novel multi-conditioned gas classification method for the first time. We present a new deep learning network structure that can efficiently extract features from the data of multiple conditions and effectively integrate them, which is referred to as a multi-conditioned gas classification network (MCGCN). We also propose a new training loss function to guarantee good performance reliably for the varying number of given conditions. Experimental results demonstrate the superiority of the proposed method, which achieves accuracies of  $99.15\% \pm 0.41$  regardless of the number of conditions with 15 times fewer model parameters in comparison to the existing method.

**INDEX TERMS** Deep learning, gas classification, data integration.

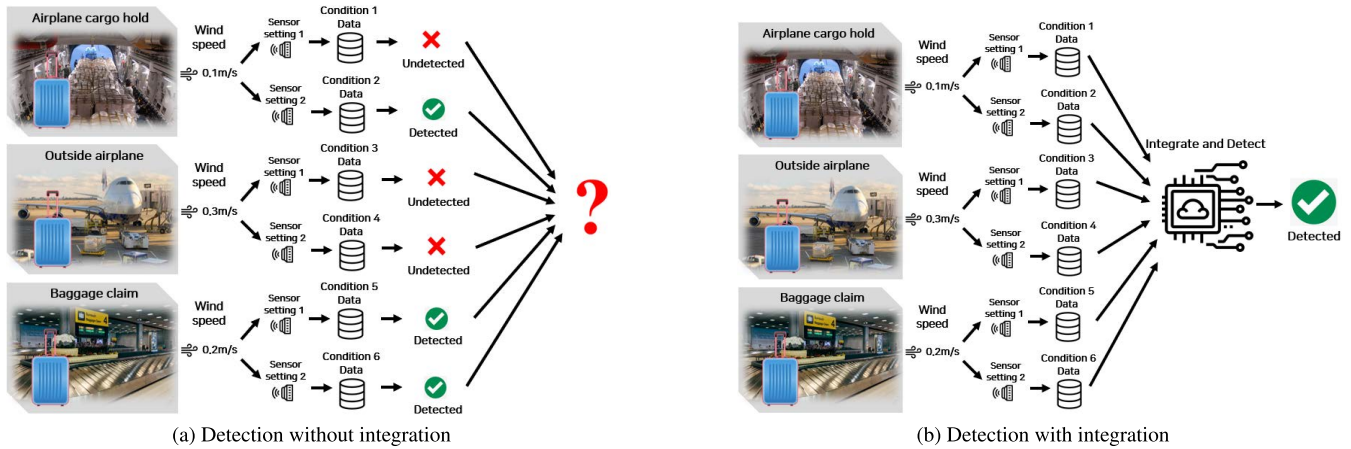
## I. INTRODUCTION

In recent years, with the development of electronic nose, new potential solutions using gas detection have been attracting attention in various areas including monitoring systems, health care, public security, and hazardous chemicals detection, to name a few [1]–[4]. The electronic nose typically consists of a sensor array that detects chemicals in the air, which is designed to mimic the biological olfactory system [5], [6]. The sensors include metal-oxide sensors, conducting polymer sensors, quartz crystal microbalance sensors, etc. [7]–[9], and distinct sensors are combined in the array to detect various chemicals. After the chemicals are captured by the sensors, the sensor signal is pre-processed to extract features, and data analysis is applied to identify the input chemicals through pattern recognition and machine

learning algorithms. In the early stage of gas classification research, traditional machine learning methods have been popularly used, such as k-nearest neighbor (kNN) [10], [11], Gaussian mixture model (GMM) [12], multi-layer perceptron (MLP) [13]–[15], and support vector machine (SVM) [16]–[20]. Recently, several attempts [21]–[24] have been made to improve the performance of gas classification by applying deep learning methods that have shown excellent performance in the computer vision and natural language processing fields.

An important issue that is specific to gas classification is the measurement environment of data, which has not been addressed much in the past. Measuring the characteristic of gas molecules is greatly affected by external factors such as wind speed due to their physical property [25], [26]. In addition, in the case of metal-oxide (MOX) sensors, which are widely used for gas detection, the device characteristic varies according to the temperature of the internal heater,

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



**FIGURE 1.** Illustration of gas classification for airplane drug smuggling detection under multiple conditions. As in (a), if the detection results are different depending on the conditions, it is difficult to decide which result to choose. Thus, it is important to properly integrate the results of the conditions as in (b).

which affects the detection result [27]. Therefore, the measured gas data tend to change depending on the external factors and the sensors' internal factors, which we call a condition in this paper. For accurate classification, a method that can classify the gas data by considering the measurement condition is required. For example, when we need to inspect a suitcase to prevent aircraft drug smuggling as shown in Figure 1, the wind speed may vary depending on the location where the gas is measured, and the measurement results may also vary depending on the sensor settings (i.e., the internal temperature of the sensor). Therefore, different gas data may be obtained even for the same target suitcase depending on the condition, which would be classified as different types of gas by existing gas classification methods that do not consider measurement conditions. In addition, since the classification results may vary depending on the condition, as shown in Figure 1(a), an additional method to integrate the different results is also required for reliable gas detection. Thus, for the deployment of a gas classification method in a real situation, classification should be done by taking into account these multiple conditions, which we call *multi-conditioned gas classification* (Figure 1(b)).

As a way of considering the measurement condition, SimResNet [28] was proposed to receive two types of input, which are gas sensor data and external factors, for improved classification performance. However, although data from individual conditions can be classified well, SimResNet is not capable of integrating data from multiple conditions. In other words, SimResNet can only handle one condition. In order to solve this problem, EmbraceNet [29] can be employed, which was designed to perform classification with integrating different types of data. For multi-conditioned gas classification, it may be possible to adopt EmbraceNet by regarding the gas sensor signals measured under different conditions as different types of data and obtaining an integrated classification result for them. However, EmbraceNet has never been used to classify

gas data from multiple conditions, and as it will be shown through our experiments, its performance is not satisfactory. In addition, EmbraceNet requires substantial computational resources depending on the number of conditions. Moreover, if there are a large number of data from different conditions, it cannot fully learn the data from individual conditions.

In this paper, we propose a multi-conditioned gas classification model for the first time to the best of our knowledge. We present a new deep learning network that can classify gas data measured under multiple conditions, which we call a multi-conditioned gas classification network (MCGCN). Considering the characteristics of gas data, we employ a shared feature extraction module (SFEM) for efficient and effective feature extraction, which achieves high performance with a significantly reduced number of parameters. Then, in order to achieve high performance regardless of the presence or absence of each condition, we adopt the method in EmbraceNet to integrate the features of the individual conditions. Furthermore, to allow the network to better learn information of data from distinct conditions, we propose a new training loss function consisting of the integrated classification loss and the losses for separate conditions. Since the integration method in EmbraceNet reduces the impact of individual conditions on the final classification result, as the number of conditions increases, it is challenging for the integrated feature to fully cover the information of each condition. However, by learning the feature information of data from individual conditions, the overall performance of MCGCN is significantly improved when compared to the method in EmbraceNet. In particular, when the number of considered conditions is very few, the performance gains are even greater.

To sum up, our contribution is summarized as follows.

- We present the first multi-conditioned gas classification model that can effectively classify gas data measured under multiple conditions, a multi-conditioned

gas classification network (MCGCN), which achieves much higher performance compared to the existing EmbraceNet [29] and SimResNet [28].

- We propose a new deep learning network using a shared feature extraction module (SFEM), which achieves high performance with a significantly fewer number of parameters.
- We propose a new learning loss function that allows the network to not only integrate the data from multiple conditions effectively but also learn the data from individual conditions better.

The rest of the paper is organized as follows. Section II provides a brief survey of the related work. Section III explains the proposed MCGCN and its training method. Experimental results are provided in Section IV. Finally, conclusion is given in Section V.

## II. RELATED WORK

### A. GAS CLASSIFICATION

In the early gas classification researches, there were many studies to use traditional machine learning methods, such as k-nearest neighbor (kNN) [10], [11], Gaussian mixture model (GMM) [12], multi-layer perceptron (MLP) [13]–[15], [30], and support vector machine (SVM) [16]–[20]. For example, in [15], an MLP was used to classify five types of gases using raw sensor data. In some of these studies, extracting useful hand-crafted features was investigated extensively, such as features extracted by principal component analysis (PCA), linear discriminant analysis (LDA), or Euclidean normalization (EN). For example, the work in [30] employed EN features for MLP-based classification of four types of gases using eight sensors.

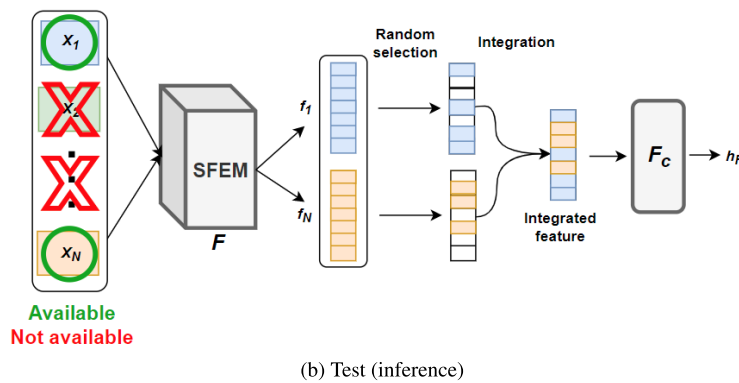
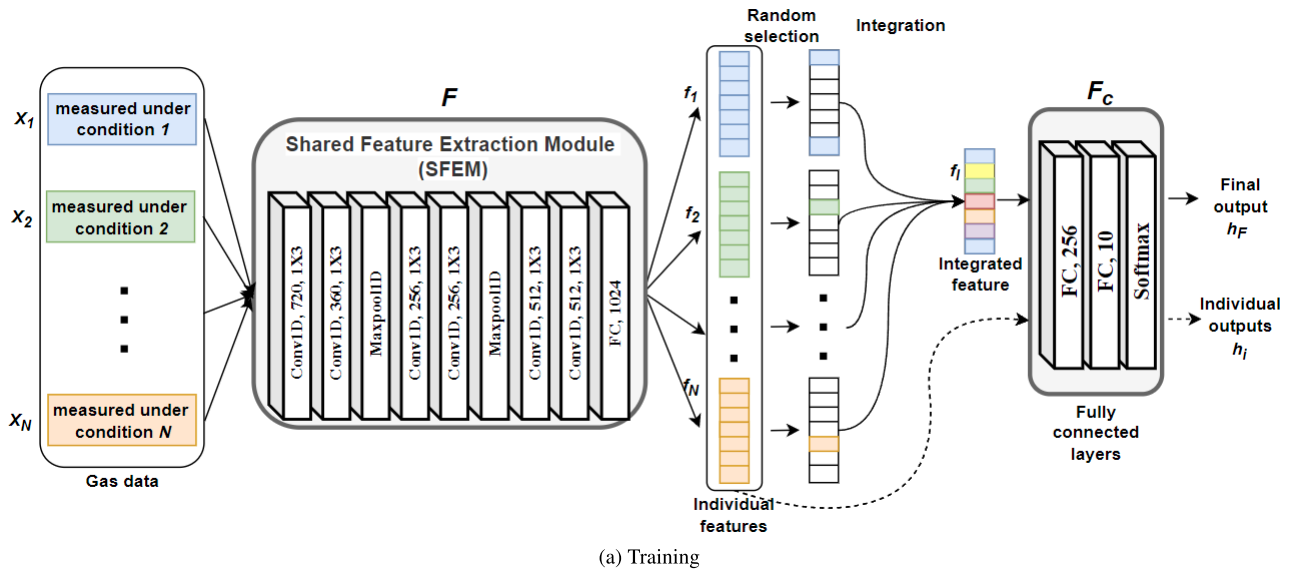
Since deep neural networks (DNNs) have shown remarkable achievements in many fields, DNN-based gas classification researches have been conducted. The work in [21] proposed a gas classification model based on a convolutional neural network (CNN), which comprises 38 layers including convolutional blocks, global average pooling layers, and fully connected layers. The studies [22], [23] solved a multi-label classification problem in mixture gases classification scenarios using a one-dimensional CNN. In [24], LeNet-5, which achieved high performance in the image classification field, was used to attain higher computation speed with less convolutional blocks.

Unlike previous researches, the work [28] considered that external factors, such as humidity, wind speed, etc., can affect the performance of gas classification and suggested a classification model, called SimResNet, which is based on ResNet [31]. Features are extracted from the external factors using an MLP, which are inputted to the model together with the gas data. By considering the external factors, SimResNet achieved high performance for data from individual conditions. However, this method did not attempt to integrate multiple results for the data from different conditions.

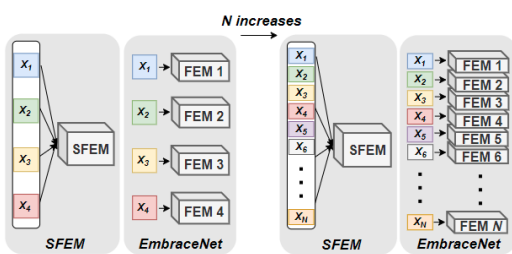
### B. INFORMATION INTEGRATION

Integrating data from multiple conditions (referred to as modalities in some studies [29], [32]–[39]) for classification arises in many fields, for which several methods have been developed [32]–[44]. A typical method is early integration (data fusion), which combines the collected data into one data and then uses it to produce the classification result. Another typical method is late integration (decision fusion), which uses each data to obtain an output and then combines the results. In [45], early integration and late integration were compared for classification of semantic concepts of videos based on visual, auditory, and textual information, where early integration showed better performance. The work in [46] classified emotion using late integration, which integrates the classification results obtained by using SVM on electroencephalogram (EEG) data and physiological signal data. Recently, early integration and late integration have also been widely implemented by the deep learning framework. The work [32] employed early integration to merge the data from multiple wearable sensors for activity recognition using CNNs and recurrent neural networks. In [40], music genre classification was conducted by integrating each classification result for hand-crafted features of the sound and a visual representation of the sound using a CNN. In addition to early and late integration, some researchers proposed intermediate integration. In [35], a bimodal deep autoencoder was suggested for speech classification using video and audio data as inputs. In this model, the feature maps of each input type are merged in the middle of the network.

However, these early, late, and intermediate integration methods are suitable for situations where the number of conditions is fixed. In many real situations, the number of conditions varies, which an integration scheme needs to be able to flexibly handle. EmbraceNet [29] integrates features of different types (e.g., kinds of gas sensors) of data by random selection and produces a single combined output. This structure was shown to be more effective than other early, late, and intermediate integration methods. Furthermore, it is robust to loss of data from certain sensors thanks to the multinomial random sampling. However, there are two limitations in using EmbraceNet directly for multi-conditioned gas classification. First, a separate network (typically a CNN) processes each type of data for feature extraction. For a large number of conditions, such feature extraction parts become significantly large and require substantial memory usage, thus it is not suitable for efficient gas classification. Second, EmbraceNet reduces the impact of individual conditions on the final classification result. Since it first integrates features of different types of data and then performs learning based on the integrated feature, as the number of conditions increases, it is challenging for the integrated feature to fully cover the information of each condition. As a result, the classification performance may become poor when only a few conditions are available. In this paper, we propose an efficient and effective model that can



**FIGURE 2.** Illustration of the proposed multi-conditioned gas classification network (MCGCN). (a) For training, gas data of all conditions are used. (b) For test, inference is performed using only the data of available conditions.



**FIGURE 3.** Comparison of SFEM and feature extraction modules (FEM) in EmbraceNet [29].

overcome these limitations. Moreover, we design a training method of our model to ensure robust performance even in the case where many (over 100) or a very few (1 or 2) conditions exist. We also show that our method achieves satisfactory performance in the real-time classification tasks.

### III. PROPOSED METHOD

In this section, we describe our proposed deep network, a multi-conditioned gas classification network (MCGCN),

and its learning algorithm for multi-conditioned gas classification. The overall framework of our proposed method is illustrated in Figure 2.

First, a new network structure, a shared feature extraction module (SFEM), extracts features from the gas data measured under different conditions. The data from each of different conditions are processed by SFEM, from which we obtain the features of the corresponding condition. In the previous work [29], separate feature extraction modules were employed for different types (conditions) of data. However, this is not suitable for multi-conditioned gas classification, and the advantages of SFEM will be explained in Section III-A in more details.

After obtaining the features of multiple conditions, we construct an integrated feature by random sampling. This approach enables the model to integrate the features automatically and to learn the data from all conditions effectively. With the integrated feature, we obtain the final output through the fully connected layers, from which the final classification loss is obtained as the cross-entropy loss.



**Algorithm 1** Multi-Conditioned Gas Classification**Input:** gas data under condition  $i$   $X_i$ , label  $Y$ **Model:** SFEM  $F$ , FC layers  $F_c$ Initialize:  $F$ ,  $F_c$ .**for**  $i = 1$  **to**  $N$  **do** $f_i = F(X_i)$ **end for****for**  $j = 1$  **to**  $f_i$ 's dimension **do** $k \sim \mathcal{U}(0, 1)$  $i = \lceil * \rceil k \times N$  $f_I^j = f_i^j$ **end for** $h_F = F_c(f_I)$ 

In addition to the final loss, the newly designed training loss function helps fully learn the data from each condition. We utilize the extracted features of each condition to obtain the individual losses and include these losses in the training loss function, which achieves significant performance boost.

In the test phase, some of the conditions used for training may not be available depending on the measurement situation. To handle this, the integrated feature is completely constructed with the data from the available conditions. Thus, our model robustly performs classification regardless of the number of available conditions.

**A. FEATURE EXTRACTION MODULE**

We denote the gas data measured under condition  $i$  as  $X_i$ , ( $i = 1, \dots, N$ ) ( $N$  is the total number of conditions) and the one hot-encoded target label as  $Y$ , where  $X_i \in \mathbb{R}^{l \times w}$  ( $l$  is the time length and  $w$  is the number of sensors) and  $Y \in \mathbb{R}^K$  ( $K$  is the number of classes). To extract the features of the gas data from each condition, we need to employ a feature extraction module, such as a CNN. To this end, we propose a shared feature extraction module (SFEM) based on a 1D-CNN. The data  $X_i$  is converted into an individual feature  $f_i$  through SFEM  $F$ , i.e.,

$$f_i = F(X_i, \theta), \quad (1)$$

where  $\theta$  is the weight parameters of the module  $F$ . Our SFEM consists of six 1D-convolutional layers with ReLU activation functions and a fully connected (FC) layer as shown in Figure 2. 1D-max pooling layers are added after the second and fourth convolutional layers to reduce the size of the feature. Note that (1) is repeated  $N$  times with  $i = 1, 2, \dots, N$ , which incurs time complexity of  $O(N)$ , and the structure of SFEM is fixed regardless of the value of  $N$ . We construct the SFEM structure based on the networks proposed in [29] and [23], which are the existing gas classification networks, and the details (the number of layers, depth, width, etc.) are determined heuristically through experiments.

There are several benefits of using SFEM over using distinct feature extraction modules for different conditions

as in EmbraceNet as shown in Figure 3. First, it can reduce the amount of memory consumed by the feature extraction module. Since EmbraceNet uses a distinct feature extraction module for each condition, memory usage increases rapidly as the number of conditions increases, which is problematic in resource-constrained environments. As will be compared in Table 2, our method can significantly reduce the memory usage compared to EmbraceNet. Second, we can increase the size of the module easily. Since the size of a module is roughly proportional to its capability to learn in most cases, using a larger module usually helps to improve performance [31]. For a fixed number of weight parameters, the size of the feature extraction module can be increased when SFEM is used instead of separate modules. Finally, it has the effect of increasing the number of training data for the feature extraction module. When a separate module is used for each condition, each module is trained only with the data for the corresponding single condition. However, when a single module is shared across all conditions as in our SFEM, the data for all the  $N$  conditions are used for training the shared module.

**B. TRAINING LOSS FUNCTION**

After extracting the features for each condition, the individual features  $f_i$  are used to form the integrated feature  $f_I$  for learning of data from multiple conditions effectively, which is inspired by [29]. The  $j$ th element of the integrated feature is assigned from the corresponding  $j$ th element of the feature of a randomly selected condition, i.e.,

$$f_I^j = f_i^j. \quad (2)$$

In this process,  $i$  is a random number drawn from the categorical distribution. Since we do not give priority to specific conditions, the probability that each condition will be chosen is set to  $1/N$ . The integrated feature  $f_I$  then goes through FC layers  $F_c$ , from which we obtain the final output  $h_F$ , i.e.,

$$h_F = F_c(f_I, \theta_c), \quad (3)$$

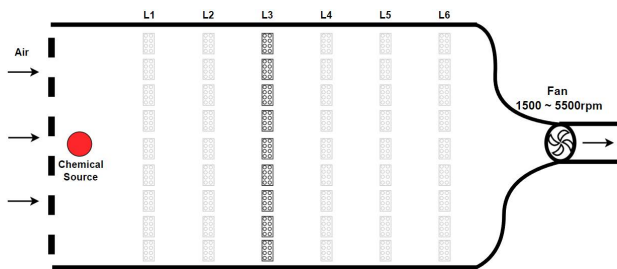
where  $\theta_c$  is the weight parameters of the FC layers.

For training our model, we can define the final loss  $L_F$  using the final output  $h_F$ , i.e.,

$$L_F = l(h_F, Y), \quad (4)$$

where  $l$  is the cross-entropy loss. However, when only the final loss  $L_F$  is used, it is difficult to fully learn the information of data from distinct conditions. First, when the integrated feature is constructed from the individual features based on random selection, data from some conditions may not be sufficiently employed. Second, when there are a large number of conditions, it is difficult to properly learn the data from each condition since the number of selected elements from each feature is very limited in the integrated feature.

Therefore, we propose a new loss function using the individual features to resolve this limitation. In computing  $h_F$



**FIGURE 4.** Wind tunnel test bed facility used to collect gas data from sensor arrays [27]. The chemical source is placed on the left side (shown as the red circle), the fan shown at the right side creates the air flow, and nine sensor arrays consisting of eight sensors are located at one of the six positions (L1 to L6).

**TABLE 1.** Class labels of mixture gas in Scenario 4. Data are divided into six classes according to the presence or absence of ethylene, methane, and CO.

Class	Ethylene	Methane	CO
1	X	X	X
2	O	X	X
3	X	O	X
4	X	X	O
5	O	O	X
6	O	X	O

(and consequently  $L_F$ ), the individual features were only used to construct the integrated feature. However, since individual features  $f_i$  have much information about the data from each condition, they can be of great help in learning the data from each condition. Thus, we compute the individual outputs using the individual features  $f_i$ , i.e.,

$$h_i = F_c(f_i, \theta_c), \tag{5}$$

and obtain the individual losses, i.e.,

$$L_i = l(h_i, Y). \tag{6}$$

Including these individual losses  $L_i$  in the training loss function helps the network learn the information of the data from the separate conditions better. Finally, the overall training loss is written as

$$L = L_F + \alpha \sum_{i=1}^N L_i, \tag{7}$$

where  $\alpha$  is a hyper-parameter to balance the final loss and the individual losses. The whole MCGCN is trained in an end-to-end manner by minimizing (7). The proposed multi-conditioned gas classification is summarized in Algorithm 1. Thus, our new training loss allows the network to learn not only the inter-related complementary information from multiple conditions, but also fully learn the distinct information from the individual conditions.

## IV. EXPERIMENTS

### A. DATASET

To evaluate the performance of the proposed MCGCN, we use the gas sensor arrays dataset [27] containing

**TABLE 2.** Comparison of separate modules and SFEM for feature extraction in EmbraceNet [29]. The last column shows the amount of memory consumed during training / inference for a given data. The proposed SFEM achieves almost the same accuracy to the case with separate modules but requires about 15 times fewer parameters and reduced memory consumption by about 50%.

Feature extraction module	Accuracy	# params	Memory
Separate modules	95.9%	393.1 M	6.2 / 2.3 GiB
SFEM	95.7%	26.4 M	3.2 / 0.8 GiB

measurements obtained from different conditions. The dataset includes various measurements for ten kinds of chemical gases: acetone, acetaldehyde, ammonia, butanol, ethylene, methane, methanol, carbon monoxide, benzene, and toluene. To collect sensor data sequences of different gases, a wind tunnel test bed facility was used, which contained the chemical source, six locations of measurement, an exhaust fan, and heating facility as shown in Figure 4. When chemical gas was exposed, data were measured by nine sensor arrays for 260 seconds with a sampling rate of 100 Hz, where each sensor array was composed of eight conductometric MOX sensors. We downsample the data to 1 Hz.

Two condition factors were considered, i.e., rotational speed of the fan (external factor) and heater temperature, which is expressed in terms of voltage, of the sensors (internal factor). The rotational speed had three options, 1500 rpm, 3900 rpm, and 5500 rpm, and the heater voltage had five options, 4.0 V, 4.5 V, 5.0 V, 5.5 V, and 6.0 V. Hence, there are 15 different conditions in total. For each condition, 20 repeated trials were conducted. Therefore, in total 18000 measurements are included in the dataset: 10 (kinds of chemical gases)  $\times$  15 (conditions)  $\times$  6 (locations)  $\times$  20 (trials). We use the data of 16 trials for training and the rest for test.

In addition, we use the mixture gas sensor arrays dataset [47] for mixture gas classification, which contains two mixture gases: ethylene and methane, and ethylene and CO. The data were measured by 16 MOX sensors (four types of sensors, four each) for 12 hours with a sampling rate of 100 Hz, which we downsample to 1 Hz. The concentration was continuously changed after a certain period of time during the measurement. Condition factors were not considered in this dataset. For each period during which the concentration is kept constant, the 30-second portion with the largest change in value is set as one batch data as in [23]. There are 694 data in total, and the data of each class label are divided into training data and test data at a ratio of 8:2.<sup>1</sup>

### B. SCENARIOS

We compare the performance of our MCGCN with that of the existing gas classification methods: EmbraceNet [29] and SimResNet [28]. In particular, we consider real situations where although data for all conditions are available in the

<sup>1</sup>We also tested a ratio of 9:1 but there was almost no difference in performance in all scenarios.

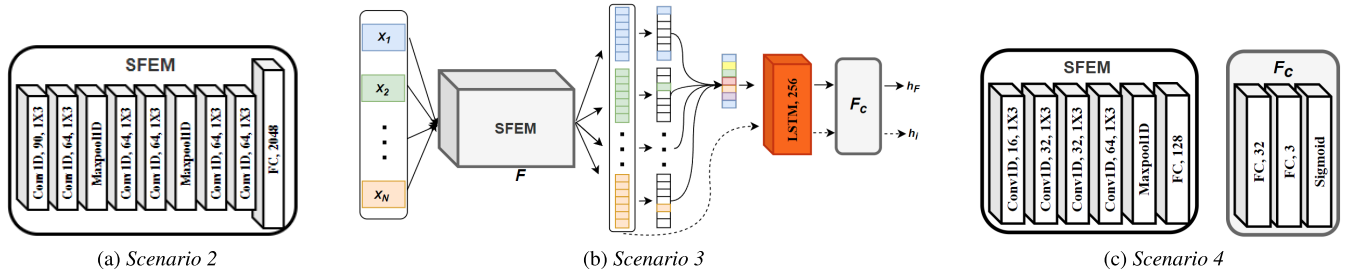


FIGURE 5. Modified parts in the network for each scenario.

TABLE 3. Effects of the individual losses in the loss function of our MCGCN in Scenario 2. When  $\alpha$  is 0, it is the same as the EmbraceNet [29] setting. The value of  $\alpha$  yielding the best performance for each number of conditions is marked in red.

Number of conditions	$\alpha$ in (7)					
	0	0.01	0.05	0.1	0.5	1.0
1	27.3%	56.5%	70.2%	68.1%	79.5%	63.1%
2	31.7%	69.8%	67.2%	76.8%	72.3%	67.0%
3	33.2%	69.3%	80.8%	88.1%	80.9%	78.0%
5	57.7%	76.8%	80.9%	84.7%	84.1%	82.8%
10	60.9%	85.8%	91.7%	92.5%	85.6%	85.6%
15	74.4%	90.3%	90.3%	91.3%	86.5%	86.7%
20	67.5%	92.5%	88.5%	92.6%	86.3%	86.9%
30	81.6%	91.2%	93.4%	93.4%	86.5%	87.5%
60	86.6%	93.5%	93.1%	93.3%	87.9%	87.2%
120	89.3%	93.8%	93.5%	93.2%	87.5%	87.7%

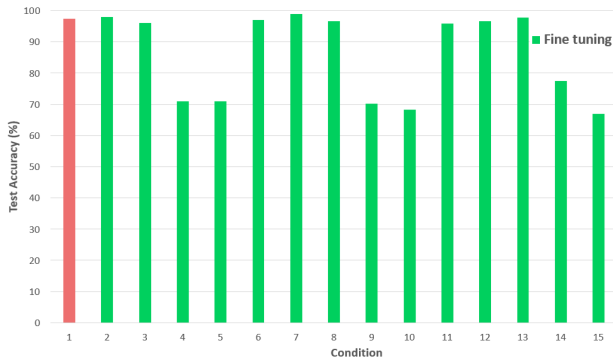


FIGURE 6. Test accuracy of the proposed model when the network is trained using the data of condition #1 and only the final FC layers are fine-tuned for each of the other conditions. The trained SFEM performs well for the conditions that were not used for training.

training phase, data for only some conditions are accessible in the test phase. Therefore, we evaluate the classification performance of the methods by changing the number of available conditions for test. The conditions available for test are randomly chosen among all possible conditions, which is repeated five times and the average accuracy and the standard deviation are reported.

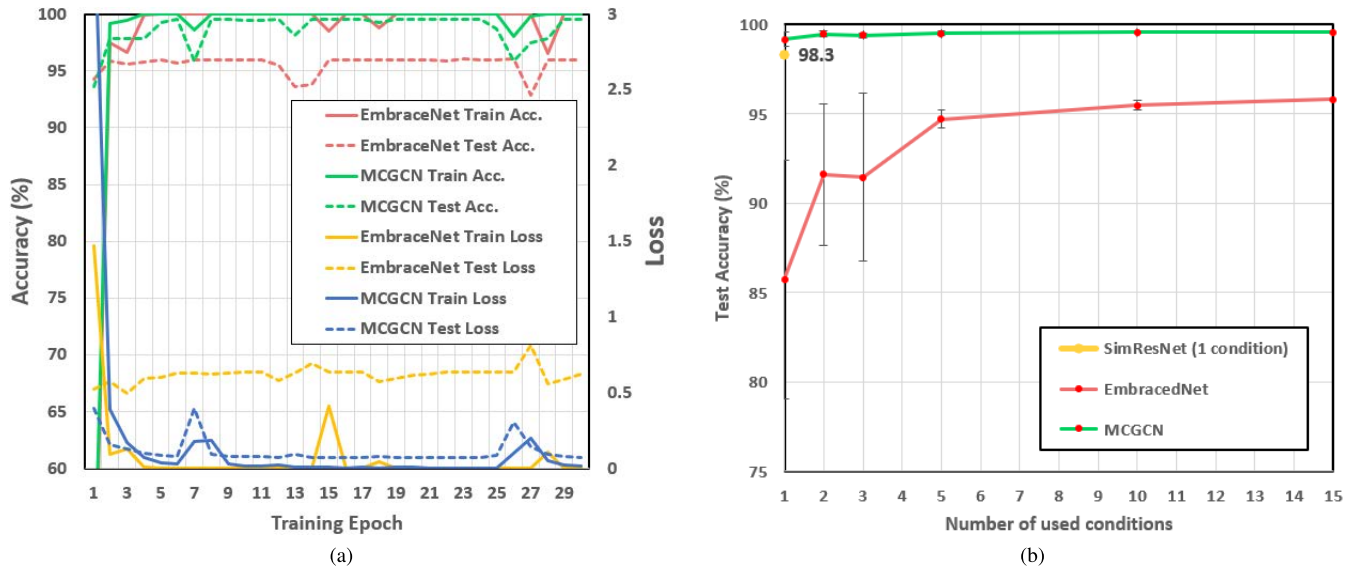
We consider three different evaluation scenarios as follows.

*Scenario 1:* We conduct an experiment with 15 conditions (combinations of five heater voltages and three wind speeds) in Section IV-E.

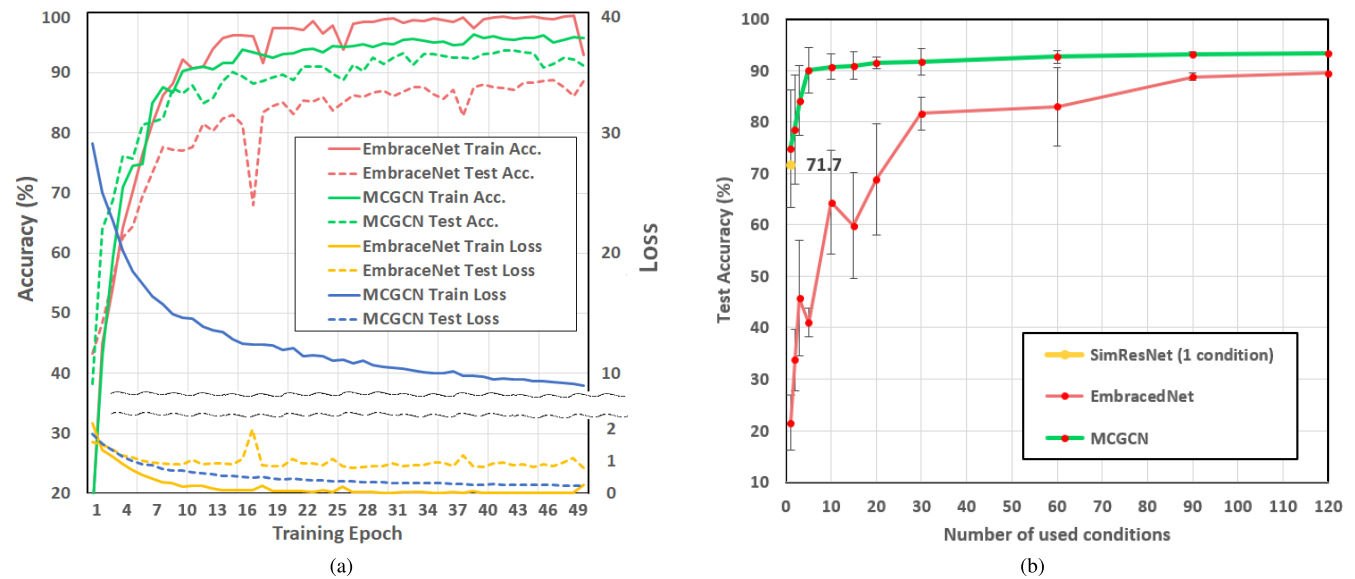
*Scenario 2:* We evaluate the performance for a more extreme case in which each kind of sensor is also considered as a condition factor in Section IV-F. Therefore, there are a total of 120 conditions (combinations of five heater voltages, three wind speeds, and eight kinds of sensors). In *Scenario 1*, 72 sensor values are considered as a whole, but here, since the eight kinds of sensors are considered as different condition factors, nine sensor values (for each type of sensor from the nine sensor arrays) are regarded as one data. In a real situation, some of the sensors may suddenly malfunction during the operation, and in this case, it is difficult to detect the gas properly if the data from all sensors are treated as one data. However, if detection is possible with only certain kinds of sensor, robust detection is possible even if some sensors fail to operate.

*Scenario 3:* We evaluate the performance for a real-time detection task in Section IV-G. In an actual deployment situation, fast detection of gas is often as important as high detection accuracy. In other words, it is necessary to achieve high performance with the data obtained for a short time period. In the previous scenarios, the classification is performed using the data for the whole time period, but in this case, the classification is performed at an interval of five or ten seconds.

*Scenario 4:* We evaluate the performance of mixture gas classification in Section IV-H. Since there are no conditions in the mixture gas dataset, the sensor type is considered as



**FIGURE 7.** (a) Learning curves of EmbraceNet [29], and the proposed MCGCN during training. (b) Test accuracy of SimResNet [28], EmbraceNet, and MCGCN with respect to the number of conditions (total 15). Our MCGCN achieves an accuracy of 99.1% on average even with only one condition.



**FIGURE 8.** (a) Learning curves of EmbraceNet [29], and the proposed MCGCN during training. (b) Test accuracy of SimResNet [28], EmbraceNet, and MCGCN with respect to the number of conditions (total 120). Our MCGCN performs better than the other methods for all cases.

a condition factor as in *Scenario 2*. Therefore, there are a total of four conditions, and four sensor values are considered as one data. This scenario corresponds to a multi-label classification problem having a total of six class labels as shown in Table 1.

**C. IMPLEMENTATION DETAILS**

We divide the time series data in the dataset using a temporal window having a length of 180 seconds and a step size of one second. Thus, the size of input data is  $15 \times 180 \times 72$ .

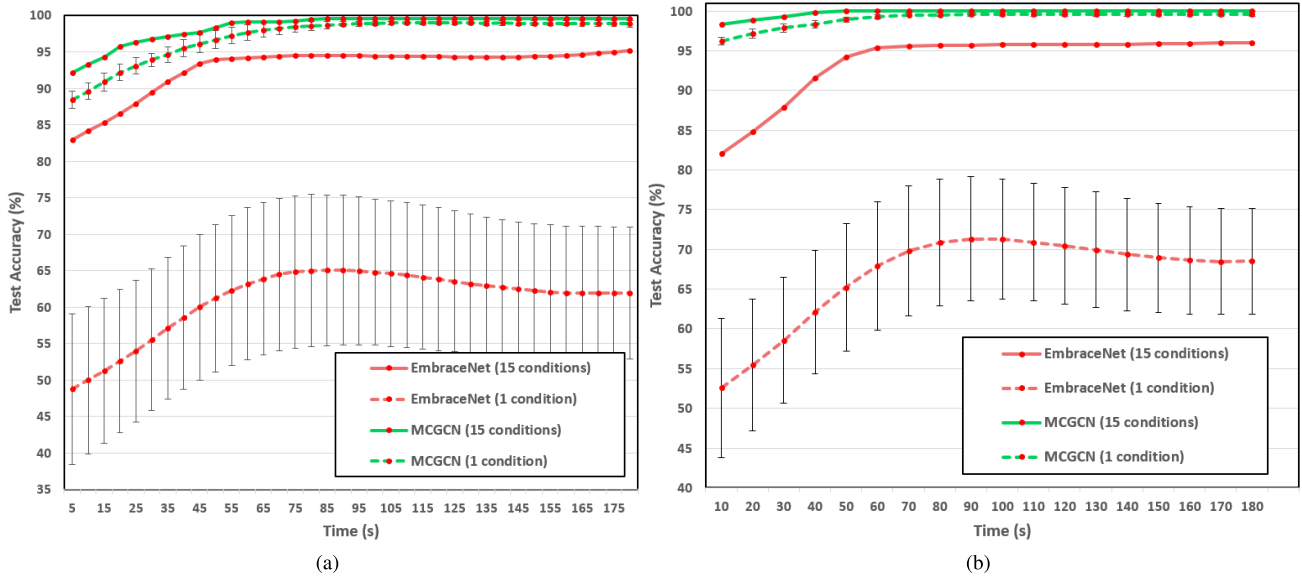
For model training, we use the Adam optimizer with a fixed learning rate of  $10^{-4}$  while the batch size is set to 80 and the

training epoch is set to 30. The size of the integrated feature is set to 1024. We set  $\alpha$  in (7) to 0.1. All experiments are implemented in TensorFlow.

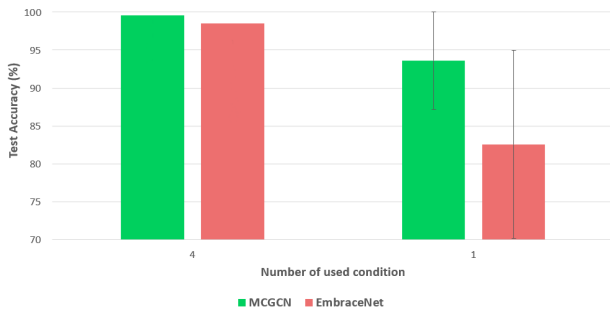
While the model structure shown in Figure 2 is used for *Scenario 1*, we slightly modify it for the other scenarios as follows.

For *Scenario 2*, since the data from 120 conditions are given as the input to the network, we need to reduce the size of the feature extraction module to avoid excessive memory consumption. Therefore, we reduce the number of channels of each convolutional layer in the feature extraction module as shown in Figure 5a. On the other hand, the dimension of





**FIGURE 9.** Test accuracy of EmbraceNet [29] and the proposed MCGCN for real-time detection when the detection sequence length is (a) 5 seconds and (b) 10 seconds. MCGCN performs significantly better than EmbraceNet for both detection sequence lengths, particularly when only one condition is available.



**FIGURE 10.** Test accuracy of EmbraceNet [29] and the proposed MCGCN for mixture gas classification. Our MCGCN outperforms EmbraceNet.

the integrated feature and the training epoch are increased from 1024 to 2048 and from 30 to 50, respectively. The reason of this is that a condition is selected among 15 conditions for each feature element in *Scenario 1*, but here, a condition is selected among 120 conditions for each element. If the dimension is set to 1024, only about 8-9 elements from each condition are reflected in the integrated feature ( $1024/120 = 8.533$ ) and it is difficult to properly reflect the information of individual conditions in the integrated feature. In addition, we increase the batch size to 120 to include as diverse conditions as possible in the learning of one iteration.

For *Scenario 3*, we add an LSTM layer having 256 units before the last FC layer in order to facilitate real-time detection by capturing temporal information better as shown in Figure 5b.

For *Scenario 4*, since the total number of data is very small and the size of each data itself is also small, the size of the entire network is greatly reduced as shown in Figure 5c. We also decrease the batch size to 20.

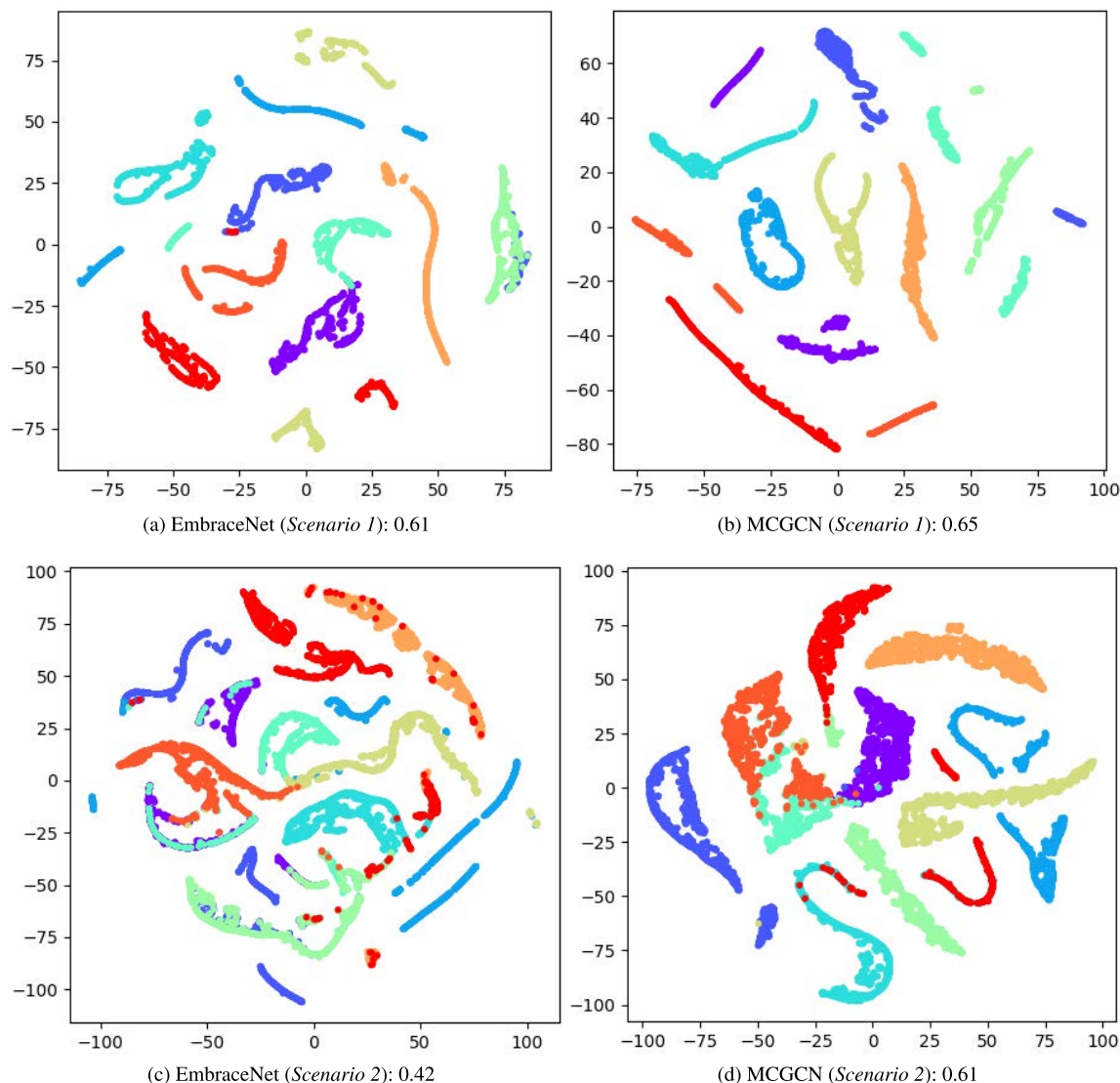
As explained in Section II-A, SimResNet [28] takes the gas sensor data and the conditions as input. The condition information processed by an MLP is merged by the intermediate integration. The model is composed of seven convolutional blocks, a global average pooling layer, a flatten layer, and two FC layers as proposed in the original work.

For EmbraceNet [29], we use SFEM for a feature extraction module as explained in Section IV-D and the rest of the parts are implemented as in the original work.

#### D. EFFICIENCY OF SFEM

In Table 2, we compare EmbraceNet with separate feature extraction modules for individual conditions and EmbraceNet with SFEM for *Scenario 1* when all 15 conditions are available. The proposed SFEM achieves almost the same accuracy as the separate modules with about 15 times fewer parameters. Since the separate modules require as many modules as the number of conditions, the number of parameters increases proportionally to the number of conditions. However, SFEM achieves efficient and effective gas classification with only one module. Moreover, even in terms of memory consumption, SFEM consumes only about a half of the memory used by the separate modules. Thus, in the following, we use SFEM as a feature extraction module in EmbraceNet for fair comparison.

SFEM can achieve high performance with a much smaller number of parameters because the data of different conditions share significant similarity and SFEM can successfully extract common features that are effective for classification. To validate this, we conduct the following experiment. First, we train the network using only the data of the first condition. Then, we apply the transfer learning technique by using



**FIGURE 11.** t-SNE (t-stochastic neighbor embedding) visualization of feature distribution in the last FC layer. Different colors indicate different classes. Normalized mutual information (NMI) scores for three-dimensional t-SNE are also reported. MCGCN produces more clearly distinct clusters corresponding to different classes.

the trained SFEM and fine-tuning only the final FC layers with the data of each of the other conditions. Figure 6 shows that even with the SFEM trained with the data of the first condition, our model can achieve significantly high performance for the other conditions. In other words, useful features can be extracted for the other conditions through the SFEM trained using the data of one condition.

**E. SCENARIO 1**

Figure 7 shows how the loss values change during training and summarizes the classification accuracy with respect to the number of available conditions for Scenario 1. The learning curves show that MCGCN successfully learns the multi-conditioned gas data and achieves higher test accuracy and lower test loss. Moreover, our MCGCN shows significant performance improvement compared to the

EmbraceNet method in all cases. In particular, it achieves high performance when there are very few conditions, e.g., 99.1% with only one condition. However, in the case of EmbraceNet, the accuracy is 85.7% with one condition, which is about 13% lower than MCGCN. The reason for this performance improvement is that MCGCN can learn the individual conditions properly because the information of each condition is directly reflected in the loss function. Conversely, it is difficult for EmbraceNet to fully learn about each condition because the individual features are only used to consist of the integrated feature. If the number of conditions increases, the performance of EmbraceNet improves to some extent, but there is still a large performance gap with MCGCN. MCGCN also performs better than SimResNet, which achieves an accuracy of 98.3%. Furthermore, SimResNet can only be used in a situation where only single

condition exists and cannot deal with the data from multiple conditions. Therefore, MCGCN is much more advantageous than SimResNet in terms of both performance and usefulness.

#### F. SCENARIO 2

Figure 8 shows the learning curves and the classification performance for *Scenario 2* where each kind of sensor is considered as a condition factor. Since MCGCN uses the additional individual losses, the training loss of MCGCN is much larger than that of EmbraceNet. However, the test loss of MCGCN is much smaller than that of EmbraceNet. In addition, the training accuracy of EmbraceNet is higher than that of MCGCN, but the test accuracy of MCGCN is higher than that of EmbraceNet. Thus, even for *Scenario 2*, MCGCN learns the multi-conditioned gas data better than EmbraceNet. In particular, if the number of available conditions is very small, the performance difference is considerably large; when only one condition is used, the difference in accuracy is about 70%. In addition, MCGCN achieves an accuracy higher than 90% with only five conditions, whereas EmbraceNet requires more than 30 conditions to exceed 90%. This shows that when there are extremely many conditions, the information of each condition is not sufficiently reflected in the integrated feature in EmbraceNet. However, since MCGCN learns by reflecting the individual features to the loss function, high performance can be obtained even with only one condition. Moreover, similar to *Scenario 1*, the performance of MCGCN is higher than SimResNet, which achieves an accuracy of 71.7%. To sum up, the proposed MCGCN is even more powerful in extreme situations with many possible conditions.

#### G. SCENARIO 3

In Figure 9, the performance of EmbraceNet and MCGCN for real-time detection is compared. In both methods, when detection is performed with only an initial portion of the data, the accuracy is somewhat low, but as time passes, the performance increases as the amount of input data increases. The performance of MCGCN is significantly higher than that of EmbraceNet regardless of the detection sequence length (5 or 10 s). In particular, if only one condition is available, a gap of about 40% or more occurs in the initial detection performance. The superiority of MCGCN shown in *Scenario 1* and *Scenario 2* also leads to satisfactory performance in real-time detection.

#### H. SCENARIO 4

In Figure 10, the performance of EmbraceNet and MCGCN for mixture gas classification is compared. As in the other scenarios, the performance of MCGCN is higher than that of EmbraceNet, and the performance difference is larger when there is only one condition. This confirms that MCGCN also performs well in multi-label mixture gas classification.

#### I. FURTHER ANALYSIS

To investigate the effect of the individual losses in the loss function, we further evaluate the performance by changing the value of the balancing parameter  $\alpha$  in (7). To this end,

we conduct experiments with the configuration in *Scenario 2*, because *Scenario 2* is relatively challenging and thus it is easier to clearly verify the effect of the individual losses. The results are shown in Table 3. In all cases, training with the individual losses achieves better performance than without them (i.e.,  $\alpha = 0$ ). When  $\alpha$  is too small ( $\alpha = 0.01$ ), the performance is poor with a small number of conditions, which means that the individual conditions are not sufficiently learned. In addition, if  $\alpha$  is too large ( $\alpha = 1$ ), the performance is low regardless of the number of available conditions. In this case, the individual losses have too much influence on the learning, so even the individual conditions are not properly learned since the learning direction becomes incoherent among the individual conditions. Therefore, the learning by the integrated feature is not performed properly, either. Overall, satisfactory performance is achieved when the value of  $\alpha$  is 0.1.

In addition, to further investigate the superiority of our method, we visualize the features of the test data with all conditions using t-SNE (t-stochastic neighbor embedding) [48] as depicted in Figure 11. The figure visualizes the feature in the last FC layer of the trained network, where each color represents each class. We also report the normalized mutual information (NMI) score [49] to quantitatively verify that the t-SNE results of MCGCN are better clustered. We apply the k-means clustering [50] to the t-SNE result and obtain the mutual information between the result of k-means clustering and the class labels. A high NMI score means that the data in the t-SNE result are clustered well according to the class labels. For obtaining NMI scores, we use the three-dimensional t-SNE results instead of two-dimensional t-SNE results in order to obtain better clustering. In both *Scenario 1* and *Scenario 2*, NMI scores are higher for MCGCN than EmbraceNet, indicating that MCGCN yields better separation between classes. We can infer that these distinct feature representations result in the high performance shown above.

#### V. CONCLUSION

In this paper, we proposed MCGCN for multi-conditioned gas classification. MCGCN consists of SFEM for efficient feature extraction and is trained with a new loss function consisting of the integrated classification loss and the losses for separate conditions, which achieves great performance enhancement compared to the existing methods. Thus, our MCGCN can classify the multi-conditioned gas data efficiently and effectively. Since there has been no method that is directly designed to classify the multi-condition gas data in previous studies, ours is considered to be an important foundation for subsequent research as a cornerstone in this field.

In the future work, we plan to extend our method to the cases with low-concentration gas data. In real situations, the concentration of chemicals in the air may be very low, so gas classification considering this is also important for practical deployment. We believe that our model will be of great help in such situations as well.

## A. LIMITATION

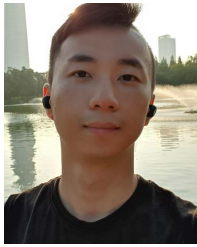
In the feature integration process, random selection can be viewed as a regularization operation, as it works similarly to a dropout function. Therefore, if the number of training data is small, it may cause underfitting. However, this problem can be alleviated by reducing the size of the SFEM.

## REFERENCES

- [1] Airsense Analytics. *New Portable Detector for Chemical Warfare Agents*. Accessed: May 8, 2021. [Online]. Available: <https://airsense.com/en/news/new-portable-detector-chemical-warfare-agents>
- [2] Electronic Nose. *E-Nose Station*. Accessed: May 8, 2021. [Online]. Available: <https://e-nose.asia/index.php/layout/layout-three>
- [3] Sensigent. *Sensigent Chemical Sensing Technology*. Accessed: May 8, 2021. [Online]. Available: <https://www.sensigent.com/chemical.html>
- [4] Figaro. *Operating Principle of Figaro Gas Sensors*. Accessed: May 8, 2021. [Online]. Available: <https://www.figarosensor.com/technicalinfo/principle/mos-type.html>
- [5] S. Ampuero and J. O. Bosset, "The electronic nose applied to dairy products: A review," *Sens. Actuators B, Chem.*, vol. 94, pp. 1–12, Aug. 2003.
- [6] S.-W. Chiu and K.-T. Tang, "Towards a chemiresistive sensor-integrated electronic nose: A review," *Sensors*, vol. 13, no. 10, pp. 14214–14247, 2013.
- [7] H. Bai and G. Shi, "Gas sensors based on conducting polymers," *Sensors*, vol. 7, no. 3, pp. 267–307, 2007.
- [8] C. Wang, L. Yin, L. Zhang, D. Xiang, and R. Gao, "Metal oxide gas sensors: Sensitivity and influencing factors," *Sensors*, vol. 10, no. 3, pp. 2088–2106, 2010.
- [9] K. Arshak, E. Moore, G. M. Lyons, J. Harris, and S. Clifford, "A review of gas sensors employed in electronic nose applications," *Sensor Rev.*, vol. 24, no. 2, pp. 181–198, 2004.
- [10] R. Gutierrez-Osuna, A. Gutierrez-Galvez, and N. Powar, "Transient response analysis for temperature-modulated chemoresistors," *Sens. Actuators B, Chem.*, vol. 93, nos. 1–3, pp. 57–66, Aug. 2003.
- [11] J. Yang, Z. Sun, and Y. Chen, "Fault detection using the clustering-kNN rule for gas sensor arrays," *Sensors*, vol. 16, no. 12, p. 2069, Dec. 2016.
- [12] S. Brahim-Belhouari, A. Bermak, M. Shi, and P. C. H. Chan, "Fast and robust gas identification system using an integrated gas sensor technology and Gaussian mixture models," *IEEE Sensors J.*, vol. 5, no. 6, pp. 1433–1444, Dec. 2005.
- [13] J. Brezmes, B. Ferreras, E. Llobet, X. Vilanova, and X. Correig, "Neural network based electronic nose for the classification of aromatic species," *Analytica Chim. Acta*, vol. 348, nos. 1–3, pp. 503–509, Aug. 1997.
- [14] S. Omatu and M. Yano, "Mixed odors classification by neural networks," in *Proc. IEEE 8th Int. Conf. Intell. Data Acquisition Adv. Comput. Systems: Technol. Appl. (IDAACS)*, Sep. 2015, pp. 171–176.
- [15] X. Zhai, A. A. S. Ali, A. Amira, and F. Bensaali, "MLP neural network based gas classification system on Zynq SoC," *IEEE Access*, vol. 4, pp. 8138–8146, 2016.
- [16] F. J. Acevedo, S. Maldonado, E. Domínguez, A. Narváez, and F. López, "Probabilistic support vector machines for multi-class alcohol identification," *Sens. Actuators B, Chem.*, vol. 122, no. 1, pp. 227–235, 2007.
- [17] Ł. Lentka, J. M. Smulko, R. Ionescu, C. G. Granqvist, and L. B. Kish, "Determination of gas mixture components using fluctuation enhanced sensing and the LS-SVM regression algorithm," *Metrol. Meas. Syst.*, vol. 22, no. 3, pp. 341–350, 2015.
- [18] Sunny, V. Kumar, V. N. Mishra, and R. R. Das, "Classification and quantification of binary mixtures of gases/odors using thick-film gas sensor array responses," *IEEE Sensors J.*, vol. 15, no. 2, pp. 1252–1260, Feb. 2014.
- [19] K. Yan and D. Zhang, "Feature selection and analysis on correlated gas sensor data with recursive feature elimination," *Sens. Actuators B, Chem.*, vol. 212, pp. 353–363, Jun. 2015.
- [20] L. Zhao, X. Li, J. Wang, P. Yao, and S. A. Akbar, "Detection of formaldehyde in mixed VOCs gases using sensor array with neural networks," *IEEE Sensors J.*, vol. 16, no. 15, pp. 6081–6086, May 2016.
- [21] P. Peng, X. Zhao, X. Pan, and W. Ye, "Gas classification using deep convolutional neural networks," *Sensors*, vol. 18, no. 2, p. 157, Jan. 2018.
- [22] Z. Wen, W. Ye, X. Zhao, and X. Pan, "A novel 1D deep convolutional neural network based algorithm for mixture gases recognition," in *Proc. IEEE 61st Int. Midwest Symp. Circuits Syst. (MWSCAS)*, Aug. 2018, pp. 893–896.
- [23] X. Zhao, Z. Wen, X. Pan, W. Ye, and A. Bermak, "Mixture gases classification based on multi-label one-dimensional deep convolutional neural network," *IEEE Access*, vol. 7, pp. 12630–12637, 2019.
- [24] G. Wei, G. Li, J. Zhao, and A. He, "Development of a LeNet-5 gas identification CNN structure for electronic noses," *Sensors*, vol. 19, no. 1, p. 217, 2019.
- [25] D. Martínez, J. Moreno, M. Tresanchez, E. Clotet, J. M. Jiménez-Soto, R. Magrans, A. Pardo, S. Marco, and J. Palacín, "Measuring gas concentration and wind intensity in a turbulent wind tunnel with a mobile robot," *J. Sensors*, vol. 2016, pp. 1–8, Jan. 2016.
- [26] L. Sun, C. Han, N. Wu, B. Wang, and Y. Wang, "High temperature gas sensing performances of silicon carbide nanosheets with an n-p conductivity transition," *RSC Adv.*, vol. 8, no. 25, pp. 13697–13707, 2018.
- [27] A. Vergara, J. Fonollosa, J. Mahiques, M. Trincavelli, N. Rulkov, and R. Huerta, "On the performance of gas sensor arrays in open sampling systems using inhibitory support vector machines," *Sens. Actuators B, Chem.*, vol. 185, pp. 462–477, Aug. 2013.
- [28] S.-H. Wang, T.-I. Chou, S.-W. Chiu, and K.-T. Tang, "Using a hybrid deep neural network for gas classification," *IEEE Sensors J.*, vol. 21, no. 5, pp. 6401–6407, Mar. 2021.
- [29] J.-H. Choi and J.-S. Lee, "EmbraceNet: A robust deep learning architecture for multimodal classification," *Inf. Fusion*, vol. 51, pp. 259–270, Nov. 2019.
- [30] F. Benrekia, M. Attari, and M. Bouhedda, "Gas sensors characterization and multilayer perceptron (MLP) hardware implementation for gas identification using a field programmable gate array (FPGA)," *Sensors*, vol. 13, no. 3, pp. 2967–2985, 2013.
- [31] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 770–778.
- [32] F. J. Ordóñez and D. Roggen, "Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition," *Sensors*, vol. 16, no. 1, p. 115, 2016.
- [33] J.-H. Choi, K. Kim, T. Park, J. Yun, J.-H. Lee, S. Kim, H. Shim, and J.-S. Lee, "Real-time integrated human activity recognition system based on multimodal user understanding," in *Proc. 25th Int. Conf. Intell. User Interfaces Companion*, 2020, pp. 89–90.
- [34] J.-H. Choi and J.-S. Lee, "Confidence-based deep multimodal fusion for activity recognition," in *Proc. ACM Int. Joint Conf. Int. Symp. Pervasive Ubiquitous Comput. Wearable Comput.*, pp. 1548–1556, 2018.
- [35] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng, "Multimodal deep learning," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2011, pp. 1–8.
- [36] M. R. Amer, T. Shields, B. Siddiquie, A. Tamrakar, A. Divakaran, and S. Chai, "Deep multimodal fusion: A hybrid approach," *Int. J. Comput. Vis.*, vol. 126, pp. 440–456, Apr. 2018.
- [37] X. Xu, Y. Bai, Y. Liu, X. Zhao, and Y. Sun, "MM-UrbanFAC: Urban functional area classification model based on multimodal machine learning," *IEEE Trans. Intell. Transp. Syst.*, early access, Jun. 7, 2021, doi: 10.1109/TITS.2021.3083486.
- [38] J. Park, M.-H. Kim, and D.-G. Choi, "Correspondence learning for deep multi-modal recognition and fraud detection," *Electronics*, vol. 10, no. 7, p. 800, Mar. 2021.
- [39] X. Chang and W. Skarbek, "Multi-modal residual perceptron network for Audio-Video emotion recognition," *Sensors*, vol. 21, no. 16, p. 5452, Aug. 2021.
- [40] Y. M. G. Costa, L. S. Oliveira, and C. N. Silla, "An evaluation of convolutional neural networks for music classification using spectrograms," *Appl. Soft Comput.*, vol. 52, pp. 28–38, Mar. 2017.
- [41] J.-H. Choi and J.-S. Lee, "Embracenet for activity: A deep multimodal fusion architecture for activity recognition," in *Proc. ACM Int. Joint Conf. Pervasive Ubiquitous Comput. Int. Symp. Wearable Comput.*, 2019, pp. 693–698.
- [42] S.-E. Moon, J.-H. Kim, S.-W. Kim, and J.-S. Lee, "Prediction of car design perception using EEG and gaze patterns," *IEEE Trans. Affect. Comput.*, vol. 12, no. 4, pp. 843–856, Oct. 2019.



- [43] J. Lee and J.-S. Lee, "Music popularity: Metrics, characteristics, and audio-based prediction," *IEEE Trans. Multimedia*, vol. 20, no. 11, pp. 3173–3182, Nov. 2018.
- [44] S.-E. Moon and J.-S. Lee, "Perceptual experience analysis for tone-mapped HDR videos based on EEG and peripheral physiological signals," *IEEE Trans. Auto. Mental Develop.*, vol. 7, no. 3, pp. 236–247, Sep. 2015.
- [45] C. G. M. Snoek, M. Worring, and A. W. M. Smeulders, "Early versus late fusion in semantic video analysis," in *Proc. 13th Annu. ACM Int. Conf. Multimedia (MULTIMEDIA)*, 2005, pp. 399–402.
- [46] G. K. Verma and U. S. Tiwary, "Multimodal fusion framework: A multiresolution approach for emotion classification and recognition from physiological signals," *NeuroImage*, vol. 102, pp. 162–172, Nov. 2014.
- [47] J. Fonollosa, S. Sheik, R. Huerta, and S. Marco, "Reservoir computing compensates slow response of chemosensor arrays exposed to fast varying gas concentrations in continuous monitoring," *Sens. Actuators B, Chem.*, vol. 215, pp. 618–629, Aug. 2015.
- [48] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, Nov. 2008.
- [49] A. F. McDaid, D. Greene, and N. Hurley, "Normalized mutual information to evaluate overlapping community finding algorithms," 2011, *arXiv:1110.2515*.
- [50] J. A. Hartigan and M. A. Wong, "Algorithm AS 136: A K-means clustering algorithm," *Appl. Statist.*, vol. 28, no. 1, pp. 100–108, 1979.



**HOJUNG LEE** received the B.S. degree in electrical and electronic engineering from Yonsei University, Seoul, South Korea, in 2016, where he is currently pursuing the Ph.D. degree with the School of Integrated Technology. His current research interests include efficient deep learning and machine learning.



**JAEHUI HWANG** received the B.S. degree from the School of Integrated Technology, Yonsei University, Seoul, South Korea, in 2019, where she is currently pursuing the Ph.D. degree. Her research interests include adversarial attack and robustness, especially on deep learning models targeting sequence data.



**HWIN DOL PARK** received the B.A. degree in computer science from the University of Minnesota, Minneapolis, MN, USA, in 2014, and the M.S. degree in computer software and engineering from the University of Science and Technology, Daejeon, South Korea, in 2017. He is currently a Researcher at the Electronics and Telecommunications Research Institute (ETRI), Daejeon. His research interests include deep learning and reinforcement learning.



**JAE HUN CHOI** received the B.S., M.S., and Ph.D. degrees in computer science from Jeonbuk National University, Jeonju, South Korea, in 1994, 1996, and 2000, respectively. Since 2000, he has been a Principal Researcher with the Intelligent Convergence Research Laboratory, Electronics and Telecommunications Research Institute (ETRI), Daejeon, South Korea. Since 2019, he has been an Assistant Professor with the School of ETRI, University of Science and Technology (UST), Daejeon. His research interests include deep and reinforcement learning, biomedical informatics, and healthcare.



**JONG-SEOK LEE** (Senior Member, IEEE) received the Ph.D. degree in electrical engineering and computer science from the Korea Advanced Institute of Science and Technology (KAIST), South Korea. He was a Researcher with the Swiss Federal Institute of Technology in Lausanne (EPFL), Switzerland. He is currently a Professor with the School of Integrated Technology, Yonsei University, South Korea. He has authored or coauthored more than 150 publications. His research interests include machine learning and multimedia signal processing. He serves as an Editor for *IEEE Communications Magazine* and *Signal Processing: Image Communication*.

...