Received February 8, 2021, accepted February 22, 2021, date of publication February 24, 2021, date of current version March 5, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3062187

An Efficient Data Augmentation Network for Out-of-Distribution Image Detection

CHENG-HUNG LIN, (Member, IEEE), CHENG-SHIAN LIN, PO-YUNG CHOU, AND CHEN-CHIEN HSU, (Senior Member, IEEE)
Department of Electrical Engineering, National Taiwan Normal University, Taipei 106, Taiwan

Corresponding author: Cheng-Shian Lin (majic0626@gmail.com)

This work was supported by the Ministry of Science and Technology, Taiwan through Pervasive Artificial Intelligence Research (PAIR) Labs under Grant MOST 109-2221-E003-024 and Grant MOST 110-2634-F-003-007.

ABSTRACT Since deep neural networks may classify out-of-distribution image data into in-distribution classes with high confidence scores, this problem may cause serious or even fatal hazards in certain applications, such as autonomous vehicles and medical diagnosis. Therefore, out-of-distribution detection (also called anomaly detection or outlier detection) of image classification has become a critical issue for the successful development of neural networks. In other words, a successful neural network needs to be able to distinguish anomalous data that is significantly different from the data used in training. In this paper, we propose an efficient data augmentation network to detect out-of-distribution image data by introducing a set of common geometric operations into training and testing images. The output predicted probabilities of the augmented data are combined by an aggregation function to provide a confidence score to distinguish between in-distribution and out-of-distribution image data. Different from other approaches that use out-of-distribution image data for training networks, we only use in-distribution image data in the proposed data augmentation network. This advantage makes our approach more practical than other approaches, and can be easily applied to various neural networks to improve security in practical applications. The experimental results show that the proposed data augmentation network outperforms the state-of-the-art approaches in various datasets. In addition, pre-training techniques can be integrated into the data augmentation network to make substantial improvements to large and complex data sets. The code is available at www.github.com/majic0626/Data-Augmentation-Network.git.

INDEX TERMS Out-of-distribution detection, image classification, anomaly detection, outlier detection, data augmentation, deep neural networks.

I. INTRODUCTION

Deep neural networks have achieved very impressive results in various computer vision tasks [1]–[3]. When training a neural network, the training data is an independent identical distribution, also called in-distribution data. On the other hand, data that does not belong to in-distribution data is called out-of-distribution data. For example in Fig. 1, traffic signs, zebra crossing, and cars are in-distribution data while birds are out-of-distribution data. When a neural network is too confident about the prediction results and gives too higher confidence scores to out-of-distribution data, these erroneous results will bring security risks to safety-critical applications such as autonomous vehicles [4], medical diagnosis [5] and sensor-fault detection for industrial safety [6], [7]. Therefore, out-of-distribution detection has become a very important research goal in artificial intelligence security issues [8].

The goal of out-of-distribution detection is to detect whether an input data comes from in-distribution or from out-of-distribution. To resolve the problem, many approaches are proposed and can be mainly categorized into three types including softmax-based approaches [9]–[11], uncertainty-based approaches [12], and generative model based approaches [13], [14]. Softmax-based approaches use the maximum value of softmax probability as a threshold to distinguish out-of-distribution data. On the other hand, uncertainty-based approaches add an additional confidence branch to provide an uncertainty score for an input. Finally,

The associate editor coordinating the review of this manuscript and approving it for publication was Tony Thomas.

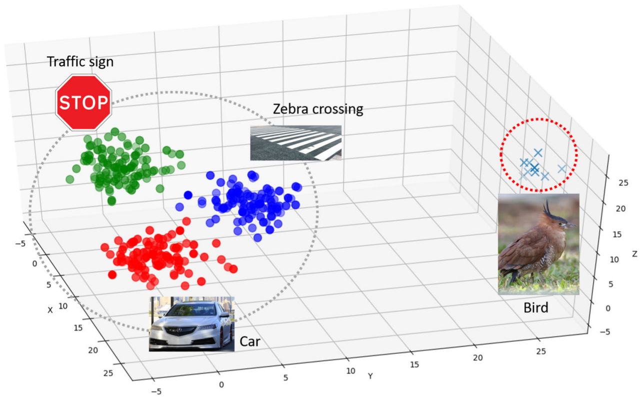


FIGURE 1. In-distribution data (gray circle) and out-of-distribution data (red circle) in feature space.

generative model based approaches treat an input as an out-of-distribution data when its corresponding output is poorly reconstructed.

Among the proposed approaches, softmax-based approaches are widely used because they can be easily combined with any neural network without modifying its original architecture or adding other models. Furthermore, they can detect out-of-distribution data without affecting the performance of primitive tasks such as classification. Therefore, many softmax-based approaches have been effectively used in pre-training models. Because softmax-based approaches use the maximum value of the softmax probability as a confidence score and compare it with a threshold, softmax-based approaches can be regarded as a binary-classification task. When the confidence score is higher than the threshold, the model predicts that the input data comes from in-distribution. Otherwise, the model predicts that the input data comes from out-of-distribution.

Although the softmax-based methods are simple and the computational cost is low, they must rely on neural networks to effectively separate the confidence scores of in-distribution data and out-of-distribution data. That is to say that a model must have the ability to give in-distribution data high confidence scores, while giving out-of-distribution data low confidence scores. However, distinguishing in-distribution and out-of-distribution data is very difficult if the confidence score is determined by only one output probability, especially for the model which is easy to be confused due to out-of-distribution data. In order to improve the accuracy of out-of-distribution detection, our idea is to introduce a set of common geometric operations into training images to generate a couple of training data. The idea comes from the assumption that data enhancement can enable the neural network to classify a set of augmented data from the same image into the same class, that is, to output similar distribution of predicted probabilities for the set of augmented images. On the contrary, when the inputs comes from out-of-distribution, the probability of obtaining a similar distribution of predicted probabilities of the augmented images is relatively small. In other words, even if one of the enhanced images has a higher predicted probability, the other probabilities will diminish its influence. Finally, these predicted probability distributions are combined by an aggregation function to obtain a confidence score, which is used to determine whether the input data comes from in-distribution or out-of-distribution.

In this paper, we develop an effective data augmentation network to detect out-of-distribution data and improve its robustness without reducing the accuracy of classification. In order to make a fair comparison, we apply the proposed method to WideResNet [15] and evaluate its effectiveness on many common datasets. The proposed data augmentation network outperforms the state-of-the-art approaches and can be further improved on larger datasets, such as TinyImageNet [16] through pre-training technique. The first innovation of this paper comes from the observation that when the input image comes from out-of-distribution, the predicted probabilities of the augmented images may be inconsistent and we make full use of this feature to detect out-of-distribution image data. The second innovation is that only in-distribution data are used to train our framework which makes our approach more practical than other approaches [10], [11], and can be easily applied to various neural networks to improve security in practical applications.

II. RELATED WORKS

A. OVERCONFIDENCE IN NEURAL NETWORKS

Neural networks have achieved significant progress on many computer vision tasks. However, we not only care about the accuracy of the model's prediction, but also how we trust the model's prediction results. For example, if the maximum value of the softmax probability distribution output by the model is 0.9, approximately 900 of the 1000 classifications performed by the neural network are correct. In other words, we can estimate how confident does the model predict for an given image by the maximum output value.

Nevertheless, neural networks are found to be overconfident occasionally for the out-of-distribution data, and classify them into a class with anomalous high scores. The MSP [9] claimed that the overconfident predictions are produced by the softmax function in neural networks because the softmax probability are computed with the fast-growing exponential function and a small addition to the softmax input will cause a large change in the output distribution. In addition, the authors in [17] also pointed out that a neural network using ReLU [18] as an activation function may output arbitrarily high confidence score to predict data that is not seen during the training phase. This problem can only be solved by changing the architecture and activation functions. In other words, a higher confidence score from the neural network does not necessarily mean that the result of the classifier is more likely to be correct, as shown in [19]. These results can be also visualized by reliability diagrams [20] which plot the gap between mean prediction accuracy and confidence scores. Surprisingly, there exists huge gaps in the modern neural networks which means that they are poorly calibrated [21], [22]. To mitigate the miscalibration of neural networks, the authors in [21] uses temperature scaling to divides the logits by T before calculating softmax values. This regularization suppresses extremely high scores in output probability while not affecting the original prediction accuracy. Moreover, multi-modal approaches are likely to reduce the over-confidence problems of deep neural networks as shown in [22].

B. OUT-OF-DISTRIBUTION DETECTION

In practical applications, deep neural networks often encounter out-of-distribution data. Due to overconfident predictions, the out-of-distribution data will seriously damage the correctness of the neural networks. To resolve the problem, many approaches [9]–[14] are proposed to detect out-of-distribution data and can be divided into three categories including softmax-based approaches, uncertainty-based approaches, and generative model-based approaches. Uncertainty-based approaches modify the architecture of neural networks to produce uncertainty score for detecting out-of-distribution data. For instance, the authors in [12] constructed an auxiliary branch onto a pre-trained classifier and derive a new out-of-distribution score from this branch. Generative model-based approaches assume that out-of-distribution data cannot be effectively reconstructed by generative model such as autoencoder or variational autoencoder. For example, the authors in [13] incorporated the Mahalanobis distance in latent space to detect out-of-distribution data by measuring reconstruction error. In [14], the authors obtained Mahalanobis distance-based score from the class conditional Gaussian distribution using hidden features in neural networks. Softmax-based approaches are widely used because of their simplicity and low computation cost. The MSP [9] proposed a baseline method using the maximum value of the softmax distribution of the classifier to detect out-of-distribution samples. Several softmax-based approaches are proposed based on this work to improve the detection performance. The ODIN [11] separated the softmax score distribution between in-distribution and out-of-distribution images using temperature scaling and adding small perturbations although fine tuning parameters for different testing data are required. Despite its low computational cost, the detection performance highly depends on the pre-trained classifier. To assist neural networks learn to differentiate between in-distribution data and out-of-distribution data, the authors in [10] proposed a method of simultaneously using Generative Adversarial Neural Networks (GAN) [23] to generate out-of-distribution data forming a boundary for in-distribution data and jointly train a classifier which should have low confidence on generated samples outside the boundary. However, training such model is computationally expensive. Moreover, tuning the hyperparameters with validation sets of out-of-distribution samples [10], [11] is often impossible since the prior of out-of-distribution samples is unavailable. Unlike only using in-distribution data in our work, the OE [24] recently proposed leveraging diverse, large real outlier images to train anomaly detectors against auxiliary datasets of outliers to improve out-of-distribution detection. Moreover, it has been shown that when neural networks are pre-trained on a large dataset such as ImageNet [25], the robustness of the model can be further improved [26] which can be integrated in our work.

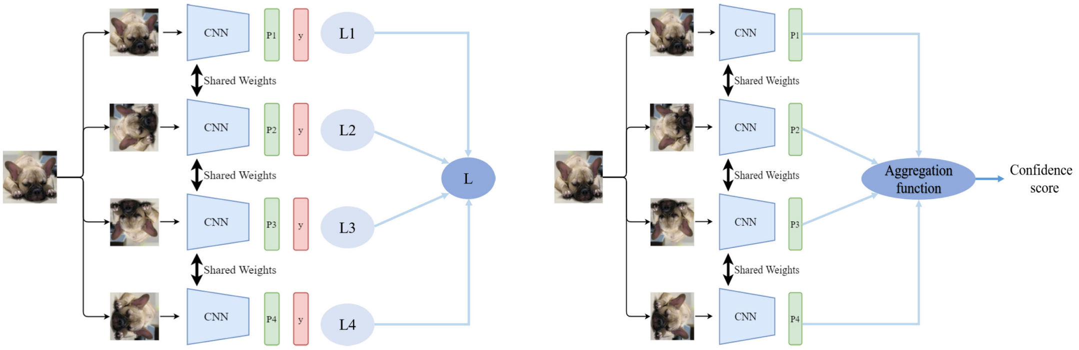
III. DATA AUGMENTATION NETWORK

In this section, we propose an efficient data augmentation network which can distinguish between out-of-distribution and in-distribution image data. There are three main components in our method including data pre-processing, data augmentation training, and aggregation function during testing phase. Fig. 2 shows the proposed data augmentation network where we introduce a set of geometric transformations, e.g. rotation, into an image to generate a set of augmented data during training and testing phases. The proposed approach requires only one CNN. When training or testing the model, the input image will be rotated into N images and sent to the CNN in turn. In the training phase, the N loss values are accumulated to the final loss which is used to update the weights of the CNN through backpropagation as shown in Fig. 2(a). In the testing phase, the input image will also be rotated into N images and sent into the trained CNN model, and then the total N predicted probabilities are aggregated to obtain the final confidence score, as shown in Fig. 2(b).

When training the model, the objective function is to classify the enhanced data from the same image into the same class. Algorithm 1 shows the training process of the proposed data augmentation network. Different from traditional training processes, the proposed network calculates the total loss after delivering the four augmented images, and then updates weights through back propagation.

After the training process, the data augmentation network will output a set of predicted probabilities for the enhanced data, and these predicted probabilities have similar distributions. We would like to mention that we only use in-distribution data to train the proposed network. This makes our method much more practical than the methods that require out-of-distribution data [10], [11], [24].

On the contrary, we assume that when input images are from out-of-distribution, the model will produce a set of predicted probabilities with inconsistent distributions. Based on this assumption, Algorithm 2 illustrates the procedure that how the model detects out-of-distribution data during the evaluation phase. Given a trained model P_θ and an input image x, a set of augmented images x_i are generated from the input image by rotation transformation $R(.)$. The model takes in enhanced data in multiple rotation angles and produces a set of predicted probabilities O_i. An aggregation function is then introduced to obtain the confidence score s from the distributions. Finally, if the confidence score is smaller than a given threshold λ, we estimate that the input image comes from out-of-distribution. In the following sections, we will



(a) Training with data augmentation

(b) Evaluation with aggregation function

FIGURE 2. (a) In the training phase, a set of enhanced data is generated from rotating input data by four angles. The model tries to classify them into the same class according to the objective function. (b) In the testing phase, a confidence score is derived from a set of predicted probabilities using aggregation functions.

```
Algorithm 1 Training Process of the Data Augmentation Network
Input:
  $P_\theta$: A model will be trained on in-distribution dataset
  $x$: Input images from in-distribution
  $R(.)$: Rotates images for $\frac{360*i}{N}$ degrees.
  $E$: Epoch for training
  $\eta$: Learning rate

  Initialize $P_\theta$
  Initialize e = 0
  for $e < E$ do                                        ▷ train the model E epochs
      $x_i = R_i(x), i = \{0, \ldots, N-1\}$            ▷ generate a set of augmented data
      $L_i = -logP_\theta(y = target|x_i)$              ▷ obtain loss for each predicted probability
      $L = \sum_{i=0}^{N-1} L_i$                        ▷ calculate final loss from each augmented images
      $P_\theta \leftarrow P_\theta + \eta \frac{\partial L}{\partial \theta}$   ▷ update weights through backpropagation
      e++
  end for

  return $P_\theta$
```

discuss the details of data augmentation, model training, and how to obtain confidence scores from aggregation functions.

A. DATA AUGMENTATION

When training neural networks for image classification, geometric transformations such as translation and rotation are often used for data augmentation [27]. However, convolutional neural networks are inherently translation-invariant, which may contradict our assumption that if out-of-distribution data is enhanced by translation, the neural network will produce inconsistent predicted probabilities.

Hence, we generate a set of enhanced images for an input image x with a set of rotation transformations $R_i, i \in \{0, .., N-1\}$, that is, each R_i rotates the image for $\dfrac{360 * i}{N}$ degrees to get $x_i = R_i(x), i \in 0, \ldots, N-1$. For example, when $N = 4$, an input image will be converted into four enhanced images by rotating the original image for 0, 90, 180, and 270 degrees. Note that x_0 is the original image without augmentation. Moreover, training images are randomly flipped and cropped in training phase to increasing data diversity.

B. MODEL TRAINING

The proposed data augmentation network can be integrated into various neural networks without modifying their architectures. Given a model P_θ and enhanced data from in-distribution images, i.e. $x \in D_{in}$, in order to classify them into the same class, the objective function is designed as (1)-(2).

$$L_{total} = \sum_{i=0}^{N-1} L_i \tag{1}$$

$$L_i = -logP_\theta(y = target|x_i) \tag{2}$$

where L_{total} denotes the sum of cross entropy for all augmented images. In other words, the model learns to classify

Algorithm 2 Testing Process of the Data Augmentation Network

```
Input:
    $P_\theta$: A model trained on in-distribution dataset
    $x$: Input images from in-distribution or out-of-distribution
    $R(.)$: Rotation transformation
    $A(.)$: Aggregation function
    $\lambda$: Threshold for detecting out-of-distribution data
    $out$: Results of out-of-distribution detection

    $x_i = R_i(x), i = \{0, \ldots, N-1\}$    ▷ generate a set of augmented data by rotating images for $\frac{360*i}{N}$ degrees
    $O_i = P_\theta(x_i)$    ▷ calculate predicted probability for each augmented images
    $s = A(O_0, \ldots, O_{N-1})$    ▷ derive a confidence score with the aggregation function
    if $s \geq \lambda$ then
        $out \leftarrow 1$
    else
        $out \leftarrow 0$
    end if

    return $out$
```

the enhanced images from the same in-distribution image into the same class as shown in Figure 2(a).

C. CONFIDENCE SCORES FROM AGGREGATION FUNCTIONS

In the testing phase, an input image x will be transformed into a group of enhanced data $R_i(x), i = \{0, \ldots, N-1\}$ as done in the training phase. The trained model then generates a set of distributions of predict probability $P_\theta(x_i), i \in 0, \ldots, N-1, P_\theta(x_i) \in R^c$, where c is number of classes. An aggregation function $A(.)$ will be introduced to derive a confidence score by combining the above distributions as shown in Figure 2(b). The following shows all candidate aggregation functions we have used in this work.

1) MEAN OF MAXIMUM VALUE (MeanMax)

Although out-of-distribution data have been statistically shown lower maximum value of softmax probability according to the baseline [9]. It also has been found that some individual out-of-distribution data lead to relatively higher confidence score. Hence, to detect out-of-distribution data accurately, we aggregate the prediction distributions from the multiple augmented images and assumes that anomalously high confidence score from the first image will be suppressed by others. The assumption will be verified in the next section. Equation (3) shows the confidence score obtained by calculating the mean of maximum value of all predicted probabilities, which is called MeanMax.

$$s = \frac{1}{N} \sum_{i=0}^{N-1} max(P_\theta(x_i)) \tag{3}$$

2) MAXIMUM OF ALL VALUES (MaxMax)

It has been shown that in-distribution data tends to produce higher confidence score than out-of-distribution data [9]. In addition to using the mean of maximum value of all predicted probabilities, we also test the effectiveness of calculating the maximum value as a confidence score in all predicted probability distributions. Equation (4) shows the confidence score obtained by calculating the maximum value of all predicted probability distributions, which is MaxMax.

$$s = max\{max(P_\theta(x_0)), \ldots, max(P_\theta(x_{N-1}))\} \tag{4}$$

3) MEAN OF POSITIONAL MAXIMUM VALUE (MeanPos)

In the training phase, a neural network learns to classify all enhanced data from an in-distribution image into the same class. We assume that the model will encounter out-of-distribution data during the testing phase and predict them as inconsistent classes. Based on the above assumptions, (5)-(6) shows the confidence score obtained by averaging predicted probability from the same index where has maximum value in predicted probability of the original image without augmentation.

$$s = \frac{1}{N} \sum_{i=0}^{N-1} P_\theta^j(x_i) \tag{5}$$

$$\arg\max_j P_\theta^j(x_0) \tag{6}$$

where $P_\theta^j(x_i)$ means the j_{th} value in predicted probability for x_i.

4) JENSEN-SHANNON DIVERGENCE (JSD)

In [10], the authors proposed the confidence loss by adding the confidence term based on Kullback-Leibler divergence (KL) on the basis of the cross-entropy loss, in which it is assumed that the predicted probability of the model should be more uniform when the data is from out-of-distribution. Therefore, we can detect out-of-distribution data by measuring the similarity between the prediction distribution and the uniform distribution. JSD [28] has been chosen in this work because its output is between 0 and 1, which clearly indicates the confidence of the prediction with proper normalization. Equation (7)-(8) show how to calculate JSD for two probability distributions, P and Q.

$$JSD(P||Q) = \frac{1}{2}KL(P||M) + \frac{1}{2}KL(Q||M) \tag{7}$$

$$KL(P||M) = \sum_{i=0}^{c-1} log_2 \frac{P_i}{Q_i} \tag{8}$$

where $M = \frac{P+Q}{2}$ and c is the number of classes.
Equation (9) derives the confidence score from JSD by setting P as output probability and Q as uniform distribution.

$$s = \frac{1}{N}\sum_{i=0}^{N-1} JSD(P_\theta(x)||U) \tag{9}$$

where $U = \frac{1}{c}(1, \ldots, 1)$, $U \in \mathbb{R}^c$, and c is the number of classes.

5) MAXIMUM VALUE IN PREDICTION PROBABILITY (MSP)

The baseline proposed the MSP as the confidence score which is also one of the aggregation functions, as shown in (10). Instead of using a set of enhanced images, they only derived the confidence score from the original image without augmentation.

$$s = max(P_\theta(x_0)) \tag{10}$$

IV. EXPERIMENTAL RESULTS

In this section, we conduct a set of experiments to evaluate the effectiveness of our data augmentation network for out-of-distribution detection. In [23], pre-training claims to improve the robustness and uncertainty of neural networks, although it is reported that it has no significant impact on the classification accuracy of the model [29]. In [23], the baseline method is re-implemented using 40-2 WideResNet for classifying CIFAR and TinyImageNet [30] datasets. To compare with their the results, we choose the above three datasets and their corresponding testing data as in-distribution samples to evaluate the effectiveness while various natural datasets including SVHN [31], LSUN [32], Texture [33], Place365 [34], and synthetic dataset such like Blob, Gaussian, Rademacher are chosen as out-of-distribution samples.

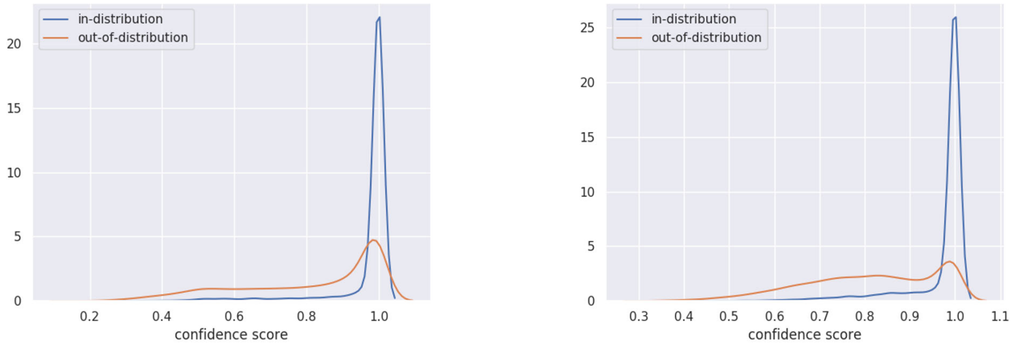
Our data augmentation network can be regarded as a threshold-based detector. If the confidence score of a given input image x is lower than the threshold λ, it will be predicted as an out-of-distribution sample. We evaluate the effectiveness of our framework with the following four metrics:

- False positive rate (FPR) at 95% true positive rate (TPR). Let TP, TN, FP, and FN denote true positive, true negative, false positive, and false negative, respectively. We evaluate FPR ($\frac{FP}{FP+TN}$) when TPR ($\frac{TP}{TP+FN}$) is 95%.
- Area Under the Receiver Operating Characteristic curve (AUROC) [35]. Receiver Operating Characteristic (ROC) curve uses varying thresholds to plot the relationship between TPR and FPR. The larger the AUROC value, the better the performance. A model is an ideal detector when its AUROC reaches 1.
- Area Under Precision-Recall curve (AUPR) [36]. Precision-Recall (PR) curve plots the relationship between Precision ($\frac{TP}{TP+FP}$) and Recall ($\frac{TP}{TP+FN}$) by varying a threshold. The larger the AUPR value, the better the performance.
- Detection error (DetErr). We evaluate the effectiveness of the detector by find the minimum classification error for all thresholds. The DetErr can be defined as $P(x_{in})P(err_{in}|x_{in}) + P(x_{out})P(err_{out}|x_{out})$. The lower the DetErr value, the better the performance. Note that err_{in} indicates that the confidence score of the in-distribution data is lower than the threshold while err_{out} indicates that the confidence score of the out-of-distribution data is higher than the threshold. We also suppose that the prior of in-distribution data $P(x_{in})$ and out-of-distribution data $P(x_{out})$ are both 0.5.

In addition, we only compare certain metrics with other works based on the metrics shown in their results, such as AUROC and AUPR. In this work, we train our model from scratch using SGD with Nesterov momentum and a cosine learning rate. The initial learning rate is set to 0.1, and it decays to 1e-6 for 100 epochs without restarting. Also, dropout is set to 0.3 to prevent overfitting. In addition, when we apply the proposed data augmentation network to a pre-trained network, the dropout will not be used and the learning rate is set to 0.01.

A. VALIDATION OF OUR ASSUMPTION

In this paper, the proposed data augmentation network is based on the assumption that when an input sample comes from out-of-distribution, the confidence score should be low. To validate our assumption, we choose in-distribution data from CIFAR-10 while out-of-distribution from Texture, SVHN, Places365, LSUN, CIFAR-100, Gaussian, Rademacher and Blob. In Fig. 3, x-axis and y-axis represent the confidence scores and the number of data in percentage, respectively. Blue lines indicate the distribution of confidence scores for in-distribution data while orange lines represent the distribution of confidence scores for out-of-distribution data. This aims to visualize the confidence of the model for its predictions. Out-of-distribution data are given lower confidence scores because the model has less confidence of them. In other words, the distribution shape of the confidence scores is uniform. On the contrary, in-distribution data are given higher confidence scores. Compared with the distribution of confidence scores for the baseline algorithm (MSP) shown in Fig. 3(a), data augmentation can



(a) original confidence scores without data augmentation

(b) confidence scores with data augmentation

FIGURE 3. (a) The original distribution of confidence score without data augmentation. (b) The distribution of confidence score with data augmentation. This figure shows that the distribution of confidence scores of out-of-distribution data (orange line) in (b) are more uniform than that in (a). This result shows that data augmentation can provide out-of-distribution data with lower confidence score.

TABLE 1. OOD detection performance with respect to number of rotations(N) on CIFAR-10. The symbol ↑ indicates that the larger the value, the better the performance, and the symbol ↓ indicates that the lower the value, the better the performance.

Metrics D_{out}	AUROC(%) ↑						AUPR(%) ↑					
	N=1 (MSP)	N=2	N=3	N=4	N=5	N=6	N=1(MSP)	N=2	N=3	N=4	N=5	N=6
Texture	87.0	84.9	89.0	84.5	**89.5**	88.4	55.9	52.0	59.8	51.4	**61.9**	58.7
SVHN	85.2	88.9	**90.2**	89.4	**90.2**	89.3	47.9	55.2	56.1	57.3	**60.5**	56.6
Places365	87.0	86.8	88.2	87.5	**89.2**	88.5	55.4	56.3	58.1	58.2	**62.3**	60.9
LSUN	91.3	93.7	90.6	**94.4**	91.8	91.0	64.9	73.2	64.7	**77.2**	69.3	66.1
CIFAR-100	85.4	86.0	87.9	86.9	**88.5**	88.1	52.0	52.8	55.4	54.2	**58.5**	58.1
Gaussian	76.8	49.0	98.1	98.3	83.3	**98.4**	29.4	14.8	83.4	**86.2**	33.5	85.8
Rademacher	78.6	**99.3**	96.9	99.0	84.5	88.6	28.7	**95.3**	73.2	90.8	35.3	41.9
Blob	98.2	99.4	99.3	99.5	**99.9**	98.1	89.5	**97.6**	96.5	**97.6**	99.5	91.8
avg	86.2	86.0	**92.5**	**92.5**	89.6	91.3	52.9	62.1	68.4	**71.6**	60.1	65.0

Metrics D_{out}	FPR(%) ↑						DetErr(%) ↑					
	N=1 (MSP)	N=2	N=3	N=4	N=5	N=6	N=1(MSP)	N=2	N=3	N=4	N=5	N=6
Texture	63.0	68.0	58.7	66.6	**54.6**	59.5	18.3	20.7	**17.6**	21.5	18.2	18.8
SVHN	72.5	65.7	62.5	60.3	**56.1**	62.6	20.0	16.4	**15.5**	17.1	16.7	17.3
Places365	64.6	63.7	59.3	58.9	**53.3**	56.8	18.6	19.2	18.5	19.1	**18.1**	18.5
LSUN	53.5	42.6	51.6	**35.0**	44.7	50.0	14.4	12.7	16.9	**12.4**	16.3	16.5
CIFAR-100	67.7	66.7	63.2	64.0	**57.5**	59.3	20.1	19.4	**18.4**	19.2	18.7	19.0
Gaussian	94.9	100	5.2	5.5	99.9	**3.2**	25.9	40.2	4.0	4.2	11.9	**3.5**
Rademacher	100	1.1	14.7	**0.2**	100	97.6	17.1	2.9	4.4	**2.2**	9.7	9.4
Blob	8.7	4.0	4.0	1.5	**0.0**	15.8	4.4	4.3	4.2	3.1	**0.5**	6.7
avg	65.6	51.5	39.9	**36.5**	58.3	50.6	17.4	17.0	12.4	**12.3**	13.8	13.7

provide out-of-distribution data more uniform confidence scores as shown by the orange line in Fig. 3(b). Therefore, neural networks benefit from our data augmentation network, which can distinguish between data from in-distribution and out-of-distribution.

B. ABLATION EXPERIMENTS

Appropriate data augmentation and aggregation functions plays an important role in our approach. Therefore, two ablation experiments are conducted with CIFAR-10 to determine the aggregation function and the number of rotation angles.

1) ROTATION ANGLES

In order to understand the influence of the rotation angles on the performance of the out-of-distribution detection, we choose different number of rotations (N) from 1 to 6 for experiments. As shown in Table 1, MSP (N = 1) has the worst performance because it uses only one prediction probability. On the other hand, as N increases, the confidence score can be obtained from more predicted probabilities. In other words, the original high confidence score may be suppressed by other predicted probabilities. This result can be regarded as a voting mechanism. Furthermore, the performance

TABLE 2. OOD detection performance with respect to different aggregation functions on CIFAR-10. The symbol ↑ indicates that the larger the value, the better the performance, and the symbol ↓ indicates that the lower the value, the better the performance.

Metrics	AUROC(%) ↑					AUPR(%) ↑				
D_{out}	MSP	MeanPos	MaxMax	MeanMax	JSD	MSP	MeanPos	MaxMax	MeanMax	JSD
Texture	82.6	83.9	84.0	84.5	**85.5**	47.0	45.4	52.3	51.4	**55.5**
SVHN	87.4	88.7	88.7	89.4	**90.5**	50.9	51.1	57.0	57.3	**61.2**
Places365	85.1	86.7	85.8	87.5	**88.5**	52.0	51.2	55.9	58.2	**64.1**
LSUN	92.5	92.8	94.2	94.4	**95.9**	68.4	64.1	75.3	77.2	**84.2**
CIFAR-100	84.6	86.4	85.3	86.9	**87.7**	49.2	48.9	52.0	54.2	**58.2**
Gaussian	96.6	94.1	**99.2**	98.3	95.4	71.8	57.9	**94.3**	86.2	61.7
Rademacher	97.7	97.3	98.9	99.0	**99.6**	82.6	74.0	91.2	90.8	**93.5**
Blob	98.4	96.1	99.8	99.5	**99.9**	90.8	69.7	98.8	97.6	**99.5**
avg	90.6	90.8	92.0	92.5	**92.9**	64.1	57.8	72.1	71.6	**72.2**
Metrics	FPR(%) ↓					DetErr(%) ↓				
D_{out}	MSP	MeanPos	MaxMax	MeanMax	JSD	MSP	MeanPos	MaxMax	MeanMax	JSD
Texture	72.4	73.7	65.0	66.6	**59.2**	22.8	21.5	21.8	21.5	**21.2**
SVHN	69.8	68.0	61.2	60.3	**51.3**	18.3	17.0	17.2	17.1	**16.6**
Places365	67.5	67.3	61.3	58.9	**50.2**	20.7	19.1	20.1	19.1	**18.7**
LSUN	49.2	51.1	37.0	35.0	**22.8**	13.6	12.5	12.0	12.4	**11.1**
CIFAR-100	70.3	70.8	66.3	64.0	**57.9**	21.2	19.2	20.8	19.2	**19.1**
Gaussian	20.8	63.7	**0.4**	5.5	35.5	5.0	6.1	**2.3**	4.2	4.3
Rademacher	6.3	3.4	0.2	0.2	**0.0**	3.4	3.2	2.2	2.2	**0.7**
Blob	7.2	31.5	**0.0**	1.5	**0.0**	4.1	5.5	1.2	3.1	**1.1**
avg	45.5	53.7	36.4	36.5	**34.6**	13.6	13.0	12.2	12.3	**11.6**

of our proposed model on out-of-distribution data is gradually improved, when N increases until 4. This result supports our initial hypothesis that when N is equal to 4, the 4 augmented samples from the original image are sufficiently different from each other, so if the input image comes from out-of-distribution, 4 inconsistent prediction probabilities can be generated. However, our method has poor results for images with symmetry, such as the texture set. We think the reason is that the 4 augmented samples of the original image are not sufficiently different from each other.

In addition, we also found that when N is greater than 4, the performance begins to decrease. We infer that when N is greater than 4, the difference between these augmented samples is not large enough, and the performance is reduced. Because the amount of computation in the training and testing phases of our proposed method increases proportionally with the increase of N, we choose N to be 4 to obtain a compromise between performance and computational cost.

2) AGGREGATION FUNCTIONS

Table 2 shows the performance of out-of-distribution detection with respect to different aggregation functions, where the bold numbers indicate the method with better performance. For most data sets, JSD is better than other aggregation functions. As a result, JSD is selected as the aggregation function in the following experiments. Also, JSD has worse performance on the Texture data set than other data sets. We infer that the symmetry in the Texture will cause the performance of the proposed method to degrade.

C. DIFFERENT DATASETS

After performing the above ablation experiments on CIFAR-10, we test our approach on more complicated datasets and compare with the baseline and the state-of-the-art approach. Table 3 shows that the proposed approach performs worse when the data set has more classes. For example, the AUROC and AUPR scores of our method on CIFAR-10 are much better than those on CIFAR-100 and TinyImageNet.

Because our method is based on softmax prediction probability, when the number of classes in the dataset increases, the predicted probability tends to be uniform. In other words, the confidence scores of in-distribution data and out-of-distribution data are easily overlapped and difficult to distinguish.

D. COMPARE WITH STATE-OF-THE-ART APPROACHES

The ablation experiments help us determine the number of rotation angles (N) and the aggregation function A(.) to be 4 and JSD respectively. The MSP [7] has created a simple and effective softmax-based approach for detecting out-of-distribution data and established a strong baseline which serves a foundation for many works. Also, it has been shown that the baseline could be further improved when a model is pre-trained on a large dataset such as ImageNet [21].

TABLE 3. Comparison with the baseline and the pre-trained model. The symbol ↑ indicates that the larger the value, the better the performance.

Dataset	Metric	AUROC(%) ↑			AUPR(%) ↑		
D_{in}	D_{out}	baseline [6]	pre-trained [20]	our+pre-trained	baseline [6]	pre-trained [21]	our+pre-trained
CIFAR-100	Texture	73.5	**79.7**	79.6	33.1	44.1	**49.2**
	SVHN	74.5	**79.6**	76.5	32.0	**48.5**	36.8
	Places365	74.1	74.6	**77.4**	34.0	34.2	**37.4**
	LSUN	70.5	70.9	**81.9**	28.7	27.7	**56.5**
	CIFAR-10	75.5	75.3	**77.9**	34.5	35.8	**37.1**
	Gaussian	48.8	**96.5**	94.3	14.6	**82.7**	59.6
	Rademacher	52.3	**98.8**	97.8	15.7	**92.5**	78.6
	Blob	85.9	**89.6**	86.0	44.9	**56.4**	38.4
	avg	69.4	83.1	**84.0**	29.7	**52.7**	49.2
TinyImageNet	Texture	68.7	72.4	**73.2**	29.5	31.8	**34.6**
	SVHN	86.6	**89.1**	83.2	53.2	**58.8**	45.7
	Places365	76.8	74.6	**77.5**	**36.8**	31.8	34.0
	LSUN	73.2	71.6	**84.5**	30.4	27.4	**44.5**
	Gaussian	49.4	67.4	**76.4**	15.2	21.1	**26.8**
	Rademacher	70.7	75.0	**85.7**	23.0	25.5	**37.0**
	Blob	**76.2**	69.5	74.3	**28.2**	23.1	25.2
	avg	71.7	74.2	**79.2**	30.9	31.4	**35.4**

TABLE 4. Comparison with the GAN-based method on CIFAR-10 (in-distribution). The symbol ↑ indicates that the larger the value, the better the performance.

Metrics D_{out}	AUROC(%) ↑		AUPR(%) ↑	
	GAN [10]	our proposed	GAN [10]	our proposed
SVHN	66.8	**72.0**	**71.3**	69.5
LSUN	75.1	**86.1**	77.1	**85.0**
TinyImageNet	72.0	**78.4**	74.7	**77.0**
avg	71.3	**78.8**	74.4	**77.2**

As a result, we integrate the pre-trained model into our method and compare with the baseline and state-of-the-art approaches.

Table 3 shows that our method is superior to the baseline. When using CIFAR-100 as the in-distribution data, the AUROC and AUPR scores increase by 21.0% and 65.6%, respectively. Compared with state-of-the-art approaches, our method can further improve the performance on highly complicated data set such like TinyImageNet, the AUROC score increases by 6.3% while the AUPR score increases by 12.7% using our method.

In addition, we also compare with a GAN-based approach [10] which is proposed to generate samples on the low-density boundary around the in-distribution data space. The original classifier is trained together with the proposed GAN model to learn to differentiate between in-distribution data and out-of-distribution data. We re-implement our framework on VGGNet [37] to compare with the GAN-based approach [10], as shown in Table 4. The experimental results show that our method is superior to the GAN-based approach in AUROC and AUPR. Moreover, the GAN-based approach tuned parameters to fit specific out-of-distribution dataset which should be difficult in real-world applications because the prior of out-of-distribution data is unknown. Finally, training a classifier jointly with GAN is computationally expensive.

V. CONCLUSION

We have proposed an efficient data augmentation network which assists neural networks to detect out-of-distribution image data. We have conducted several preliminary experiments to validate our assumption where parameters and aggregation functions are determined by ablation study. The experimental results show that the proposed data augmentation network achieves significant progress in out-of-distribution detection on various visual datasets. In addition, when the model has been pre-trained on ImageNet, the effectiveness of the proposed framework can be further improved. However, our method does not have good results for images with symmetry, such as the Texture set. We think the reason is that the 4 augmented samples of the original image are not sufficiently different from each other. Therefore, future work will focus on solving the problem and apply our approach in other computer vision tasks such as object detection and semantic segmentation. In addition, the proposed framework will provide effective anomaly detection on real-word applications where safety is considered the priority.

ACKNOWLEDGMENT

The authors are grateful to the National Center for High-Performance Computing for computer time and facilities to conduct this research.

REFERENCES

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2012, pp. 1097–1105.

[2] Z.-Q. Zhao, P. Zheng, S.-T. Xu, and X. Wu, "Object detection with deep learning: A review," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 11, pp. 3212–3232, Nov. 2019.

[3] W. Rawat and Z. Wang, "Deep convolutional neural networks for image classification: A comprehensive review," *Neural Comput.*, vol. 29, no. 9, pp. 2352–2449, Sep. 2017.

[4] S. Grigorescu, B. Trasnea, T. Cocias, and G. Macesanu, "A survey of deep learning techniques for autonomous driving," *J. Field Robot.*, vol. 37, no. 3, pp. 362–386, Apr. 2020.

[5] M. Jamshidi, A. Lalbakhsh, J. Talla, Z. Peroutka, F. Hadjilooei, P. Lalbakhsh, M. Jamshidi, L. La Spada, M. Mirmozafari, M. Dehghani, A. Sabet, S. Roshani, S. Roshani, N. Bayat-Makou, B. Mohamadzade, Z. Malek, A. Jamshidi, S. Kiani, H. Hashemi-Dezaki, and W. Mohyuddin, "Artificial intelligence and COVID-19: Deep learning approaches for diagnosis and treatment," *IEEE Access*, vol. 8, pp. 109581–109595, 2020.

[6] S. Mandal, B. Santhi, S. Sridhar, K. Vinolia, and P. Swaminathan, "Nuclear power plant thermocouple sensor-fault detection and classification using deep learning and generalized likelihood ratio test," *IEEE Trans. Nucl. Sci.*, vol. 64, no. 6, pp. 1526–1534, Jun. 2017.

[7] H. Darvishi, D. Ciuonzo, E. R. Eide, and P. S. Rossi, "Sensor-fault detection, isolation and accommodation for digital twins via modular data-driven architecture," *IEEE Sensors J.*, vol. 21, no. 4, pp. 4827–4838, Oct. 2021.

[8] S. Bulusu, B. Kailkhura, B. Li, P. K. Varshney, and D. Song, "Anomalous example detection in deep learning: A survey," *IEEE Access*, vol. 8, pp. 132330–132347, 2020.

[9] D. Hendrycks and K. Gimpel, "A baseline for detecting misclassified and out-of-distribution examples in neural networks," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2017, pp. 1–12.

[10] K. Lee, H. Lee, K. Lee, and J. Shin, "Training confidence-calibrated classifiers for detecting out-of-distribution samples," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2018, pp. 1–16.

[11] S. Liang, Y. Li, and R. Srikant, "Enhancing the reliability of out-of-distribution image detection in neural networks," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2018, pp. 1–15.

[12] T. DeVries and G. W. Taylor, "Learning confidence for Out-of-Distribution detection in neural networks," 2018, *arXiv:1802.04865*. [Online]. Available: http://arxiv.org/abs/1802.04865

[13] T. Denouden, R. Salay, K. Czarnecki, V. Abdelzad, B. Phan, and S. Vernekar, "Improving reconstruction autoencoder out-of-distribution detection with Mahalanobis distance," 2018, *arXiv:1812.02765*. [Online]. Available: http://arxiv.org/abs/1812.02765

[14] K. Lee, K. Lee, H. Lee, and J. Shin, "A simple unified framework for detecting out-of-distribution samples and adversarial attacks," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 31, 2018, pp. 7167–7177.

[15] S. Zagoruyko and N. Komodakis, "Wide residual networks," in *Proc. Brit. Mach. Vis. Conf.*, Sep. 2016, pp. 87.1–87.12.

[16] H. Pouransari and S. Ghili, "Tiny imagenet visual recognition challenge," Stanford Univ., Stanford, CA, USA, Tech. Rep. CS231, 2014.

[17] M. Hein, M. Andriushchenko, and J. Bitterwolf, "Why ReLU networks yield high-confidence predictions far away from the training data and how to mitigate the problem," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 41–50.

[18] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2010, pp. 807–814.

[19] H. Jiang, B. Kim, M. Y. Guan, and M. Gupta, "To trust or not to trust a classifier," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2018, pp. 5546–5557.

[20] J. Bröcker and L. A. Smith, "Increasing the reliability of reliability diagrams," *Weather Forecasting*, vol. 22, no. 3, pp. 651–661, Jun. 2007.

[21] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," in *Proc. Int. Conf. Mach. Learn. (ICML)*, vol. 70, 2017, pp. 1321–1330.

[22] G. Aceto, D. Ciuonzo, A. Montieri, and A. Pescapé, "Toward effective mobile encrypted traffic classification through deep learning," *Neurocomputing*, vol. 409, pp. 306–315, Oct. 2020.

[23] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 27, 2014, pp. 2672–2680.

[24] D. Hendrycks, M. Mazeika, and T. Dietterich, "Deep anomaly detection with outlier exposure," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2019, pp. 1–18.

[25] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.

[26] D. Hendrycks, K. Lee, and M. Mazeika, "Using pre-training can improve model robustness and uncertainty," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2019, pp. 2712–2721.

[27] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *J. Big Data*, vol. 6, no. 1, p. 60, Jul. 2019.

[28] J. Lin, "Divergence measures based on the Shannon entropy," *IEEE Trans. Inf. Theory*, vol. 37, no. 1, pp. 145–151, Jan. 1991.

[29] K. He, R. Girshick, and P. Dollar, "Rethinking ImageNet pre-training," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 4917–4926.

[30] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," M.S. thesis, Univ. Toronto, Toronto, ON, Canada, 2009.

[31] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng, "Reading digits in natural images with unsupervised feature learning," in *Proc. NIPS Workshop Deep Learn. Unsupervised Feature Learn.*, 2011.

[32] F. Yu, A. Seff, Y. Zhang, S. Song, T. Funkhouser, and J. Xiao, "LSUN: Construction of a large-scale image dataset using deep learning with humans in the loop," 2015, *arXiv:1506.03365*. [Online]. Available: http://arxiv.org/abs/1506.03365

[33] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, and A. Vedaldi, "Describing textures in the wild," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 3606–3613.

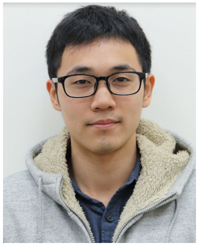
[34] B. Zhou, A. Lapedriza, A. Torralba, and A. Oliva, "Places: An image database for deep scene understanding," *J. Vis.*, vol. 17, no. 10, p. 296, 2016.

[35] J. Davis and M. Goadrich, "The relationship between Precision-Recall and ROC curves," in *Proc. Int. Conf. Mach. Learn. (ICML)*, vol. 6, Jun. 2006, pp. 233–240.

[36] T. Saito and M. Rehmsmeier, "The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets," *PLoS ONE*, vol. 10, no. 3, Mar. 2015, Art. no. e0118432.

[37] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2015, pp. 1–14.

CHENG-HUNG LIN (Member, IEEE) received the Ph.D. degree in computer science from National Tsing Hua University, Taiwan, in 2008. He is currently an Associate Professor and the Chairman of the Department of Electrical Engineering, National Taiwan Normal University, Taipei, Taiwan. He is involved in open-source software development. The multiple string matching library developed by him and his team has been widely used. His current research interests include deep learning on action recognition, auto license plate recognition, AIOT, and parallel computing.



CHENG-SHIAN LIN was born in Taipei, Taiwan, in 1994. He received the B.S. degree in electrical engineering from National Taiwan Normal University, Taipei, in 2016, where he is currently pursuing the M.S. degree. His current research interest includes anomaly detection in machine learning. He was a recipient of the Best Paper Runner Up Award from ICPAI 2020.

PO-YUNG CHOU was born in Taipei, Taiwan, in 1995. He received the B.S. degree in electrical engineering from National Taiwan Normal University, Taipei, in 2019, where he is currently pursuing the M.S. degree. His current research interests include video analysis and action recognition. He was a recipient of the Best Paper Runner Up Award from ICPAI 2020.

CHEN-CHIEN HSU (Senior Member, IEEE) was born in Hsinchu, Taiwan. He received the B.S. degree in electronic engineering from the National Taiwan University of Science and Technology, Taipei, Taiwan, in 1987, the M.S. degree in control engineering from National Chiao Tung University, Hsinchu, in 1989, and the Ph.D. degree from the School of Microelectronic Engineering, Griffith University, Brisbane, Australia, in 1997. He was a System Engineer with IBM Corporation, Taipei, for three years, where he was responsible for information systems planning and application development, before commencing his Ph.D. studies. In 1997, he joined the Department of Electronic Engineering, St. John's University, Taipei, as an Assistant Professor, and was appointed as an Associate Professor, in 2004. From 2006 to 2009, he was with the Department of Electrical Engineering, Tamkang University, Taipei. He is currently a Professor with the Department of Electrical Engineering, National Taiwan Normal University, Taipei. He has authored or coauthored more than 200 refereed journals and conference papers. His current research interests include digital control systems, evolutionary computation, vision-based measuring systems, sensor applications, and mobile robot navigation. He is a Fellow of IET.