A Dual-threshold Based Evidential Openmax Approach for Open Set Recognition

Zhuo Yang, Deqiang Han
School of Automation Science and Engineering
Xi'an Jiaotong University
Xi'an, Shaanxi, China
15202997366@163.com; deqhan@xjtu.edu.cn

Yi Yang
SKLSVMS, School of Aerospace
Xi'an Jiaotong University
Xi'an, Shaanxi, China
jiafeiyy@xjtu.edu.cn

Jean Dezert
The French Aerospace Lab - ONERA
Chemin de la Hunière
F-91761 Palaiseau, France
jean.dezert@onera.fr

***Abstract*—Traditional pattern recognition systems, tasked with categorizing inputs into known classes, often struggle when they encounter samples they haven't been trained to recognize. This introduces the need for the open set recognition—enhancing models to reject unidentified samples effectively. The Openmax method represents a significant breakthrough in this field by leveraging deep learning to spot and handle these new, unseen classes, broadening the traditional Softmax layer to accommodate an "unknown" class and employing a single threshold to separate the known from the unknown. However, the reliance of the original Openmax method on a single threshold may result in incorrect classifications if the parameters are not selected appropriately. To address this, we introduce a dual-threshold fusion mechanism based on Dempster-Shafer evidence theory in this paper. This approach releases the difficulty of finding a precise threshold in the complex and dynamic real-world environments. By integrating deep networks with a novel evidence-based system, the refined approach can bolster the robustness of rejecting undefined classes.**

***Index Terms*—Open Set Recognition (OSR), Openmax, Dempster-Shafer Evidence Theory (DST)**

I. Introduction

In traditional pattern recognition tasks, input samples are categorized into a finite set of predefined classes, assuming that all testing classes are known while training. However, when these systems are implemented in real-world settings, they usually encounter samples from classes that were not present during the training phase, termed as "unknown unknowns [1]". These classes, absent in the training data, present a significant challenge for standard classifiers during inference, as they are not equipped to recognize these novel classes. Addressing this issue becomes imperative, necessitating enhancements in recognition models to effectively reject samples from these unidentified classes. This challenge, where the goal is to discern test samples as either belonging to known or unknown classes while accurately classifying all familiar classes, is defined as the Open Set Recognition (OSR [2]).

In the field of open set recognition with non-deep learning strategies, significant attention has been directed towards the use of Support Vector Machines (SVM), especially the one-class SVM [3]. They have not only improved SVM decision boundaries to better balance empirical risks in open set scenarios, but also incorporated extreme value theory [4] using Weibull distribution. A novel adaptation, the W-SVM [5], further innovates by applying Weibull distribution to adjust output scores of binary SVMs, enhancing their effectiveness for open set recognition. Additionally, there are open set recognition methods based on nearest neighbors [6] and sparse representation [7].

The Openmax method [8], devised by Bendale and Boult in 2016, marks a significant advancement in tackling the issue of open set recognition, notably as the first initiative to integrate deep learning networks into this domain. Its primary goal is to identify and manage new classes that were not encountered during the training phase, serving as a foundational technique that has spurred a wealth of subsequent research [9]–[11]. For a closed-set recognition problem with N classes, Openmax enhances the capability of deep networks by extending the traditional Softmax layer to recognize an additional "unknown" class, thereby accommodating $N+1$ classes.

Openmax method intricately models the distribution of activation vectors for each class by analyzing the activations just before the Softmax layer and utilizing their statistical characteristics to estimate each class's distribution. Specifically, for each training class, this method applies Weibull distribution to model the tail ends of the Euclidean distances that measure how far each activation vector deviates from the class's average activation vector. Openmax evaluates the similarity between an input test sample and each trained class, generating a revised activation vector that includes $N+1$ classes. In terms of identifying unknown class samples, Openmax employs a single threshold: scores below this threshold are classified as unknown, whereas those above are deemed known. However, the area around the threshold remains somewhat unclear, potentially introducing ambiguity regarding OSR problem. This aspect of Openmax, where it doesn't explicitly articulate the uncertainty around this borderline region and relies on a single threshold, could inadvertently lead to misclassifications of certain samples.

Therefore, our proposed method in this paper incorporates a dual-threshold fusion mechanism for enhancement based on Dempster-Shafer evidence Theory (DST [12]). Building upon the Openmax framework, our work introduces DST,

This work was supported by National Natural Science Foundation of China, No. U22A2045.

which serves as an effective tool for modeling and reasoning under uncertainty, to articulate the uncertainties surrounding the decision threshold. It adopts a dual-threshold mechanism to generate Basic Belief Assignments (BBAs), which allows for the creation of distinct mass for the known classes, unknown classes, and the ambiguous area. Our approach employs various Deep Neural Networks (DNNs) as the backbone to generate distinct BBAs. Subsequently, we combine these BBAs to reduce the uncertainties to expect a better decision on the unknown classes. Three experimental series, each grounded in three different metrics, showcase the proposed method's overall enhancement over the unfused original Openmax technique.

II. Related Work

A. Openmax Algorithm

In the realm of closed-set recognition with N classes, Openmax unfolds in two distinct phases: During the initial training phase, a deep neural network featuring a Softmax layer undergoes training through the minimization of the cross-entropy loss function. The subsequent section will introduce several quintessential pattern recognition DNNs. For the adeptly trained feature extraction network, activation vectors ($\boldsymbol{AV}(x_{i,j}) \in \mathbb{R}^{N\times 1}, abbr\boldsymbol{AV}^i_j$), representing the outputs from the last fully connected layer for the sample $x_{i,j}$, the i-th sample from the j-th class of the training dataset, serve as the input for the Openmax algorithm. Openmax adopts the nearest class mean concept to depict each class by a Mean Activation Vector (MAV, $\boldsymbol{\mu}_j \in \mathbb{R}^{N\times 1}$) as illustrated below:

$$\boldsymbol{\mu}_j = \frac{1}{N_j}\sum_{i=1}^{N_j} \boldsymbol{AV}^i_j \tag{1}$$

Here, N_j and $\boldsymbol{\mu}_j$ signify the number of samples and the mean AV of images in the j-th class, respectively. For each class's training samples, the mean of the activation vectors (AV) from accurately classified known samples is computed to represent that class's MAV. This is followed by calculating the Euclidean distance between the train sample's activation vector and each class's MAV to establish a set of distances:

$$\begin{aligned} \boldsymbol{d}(i) &= \text{Distance}\left(\boldsymbol{AV}^i_j \in \mathbb{R}^{N\times 1}, \boldsymbol{\mu}_j \in \mathbb{R}^{N\times 1}\right) \\ & i = 1, 2, \ldots, N_j. \end{aligned} \tag{2}$$

These distance sets for each class are then modeled using the Weibull distribution through the libMR FitHigh function [13], deriving the Cumulative Distribution Function (CDF) for each class.

In the testing phase, Openmax essentially recalculates a new activation vector ($\widehat{AV}(x) \in \mathbb{R}^{N+1\times 1}, = \widehat{av}_1(x)...\widehat{av}_{N+1}(x)$) for a new sample x, incorporating an additional element to denote the open-set class. Its activation vector ($AV(x) \in \mathbb{R}^{N\times 1}, = av_1(x)...av_N(x)$) is selected and N_α (a hyperparameter smaller than N) of its largest elements are adjusted using two positive factors, ω_{score} and α. Openmax computes the Euclidean distance between $AV(x)$ and $\boldsymbol{\mu}_k$, the MAV corresponding to x's training class:

$$\boldsymbol{D}(x) = \text{Distance}(AV(x) \in \mathbb{R}^{N\times 1}, \boldsymbol{\mu}_k \in \mathbb{R}^{N\times 1}) \tag{3}$$

Applying the Weibull CDF to this $\boldsymbol{D}(x)$ yields ω_{score}:

$$\omega_{score} = CDF_k(\boldsymbol{D}(x)) \tag{4}$$

The Alpha rank, α, determined by N_α, is outlined as:

$$\alpha \in \left[1, \frac{N_\alpha - 1}{N_\alpha}, \frac{N_\alpha - 2}{N_\alpha}, \ldots, \frac{1}{N_\alpha}\right] \in \mathbb{R}^{N_\alpha \times 1} \tag{5}$$

Openmax modifies $AV(x)$ to $\widehat{AV}(x)$ by applying:

$$\begin{aligned} \widehat{av}_i(x) &= av_i(x) \times (1 - \omega_{score} \times \alpha_i) \\ & i = 1, 2, \ldots, N_\alpha. \end{aligned} \tag{6}$$

The deducted values are collectively added and appended at the end of $\widehat{AV}(x)$ as a new score for the open set:

$$\widehat{av}_{N+1}(x) = 1 - \sum_{j=1}^{N}(av_j(x) - \widehat{av}_j(x)) \tag{7}$$

In the end, the Softmax function together with a specified rejection ϵ threshold is applied to $\widehat{AV}(x)$:

$$\begin{aligned} \hat{P}(y = j|x) &= \frac{e^{\widehat{av}_j(x)}}{\sum_{j=1}^{N+1} e^{\widehat{av}_j(x)}} \\ y^* &= argmax_j \hat{P}(y = j|x) \end{aligned} \tag{8}$$

Openmax rejects input with two criteria:

$$y^* = N + 1 \tag{9}$$

or

$$\hat{P}(y = y^*|x) < \epsilon \tag{10}$$

In criteria Eq.(10), the Openmax approach rejects input when the maximum posterior probability $\hat{P}(y = y^*|x)$ falls below ϵ, irrespective of the associated class y^*. Such reliance on a single threshold may result in misclassifications due to rigid demarcation. A threshold that is either too high or too low can lead to varying outcomes for the same sample in OSR, highlighting an area of the ambiguity around the threshold.

B. Deep Neural Networks for Activation Vector Extraction

Openmax leverage iconic DNN architectures from the domain of machine vision for the extraction of activation vectors. The DNN models adopted in our study include VGG, GoogleNet, ResNet, DenseNet, and ResNext. Here's a concise overview:

VGG (Visual Geometry Group [14]) innovates with "inception modules" for multi-scale convolution within a single module, achieving efficiency through reduced parameters without compromising effectiveness.

GoogLeNet [15] features the innovative "Inception module" that optimizes computational resources by structuring a network within a network. This design allows for efficient data processing across different scales, maintaining high performance while minimizing computational overhead.

ResNet (Residual Networks [16]) introduces skip connections in its residual blocks to train deeper networks effectively by overcoming the vanishing gradient issue, thereby maintaining performance despite the increased depth.

DenseNet (Densely Connected Convolutional Networks [17]) enhances the feature reuse across all layers in a feed-forward manner, significantly boosting efficiency and reducing overfitting by promoting feature propagation and reuse.

ResNeXt [18] adds "cardinality" to ResNet, offering multiple parallel paths to increase capacity more effectively than simply deepening or widening the network, optimizing complexity.

Each of these networks has made significant contributions to the field of deep learning, offering various architectural innovations that enhance performance on a wide range of tasks, including open set recognition when used as part of an Openmax framework. By employing these networks as its backbone, Openmax has the capability to capture activation vectors from samples through diverse approaches.

III. Enhancing Openmax with a Dual-Threshold Approach Based on Evidence Theory

As discussed in Section II, Openmax does not directly address the uncertainties associated with threshold decisions, yet different DNNs are capable of capturing diverse activation, holding the potential to enhance OSR performance through fusion. To tackle these issues, we incorporate the DST, a robust framework designed for uncertainty modeling and the fusion of multiple sources, thereby augmenting Openmax. Our approach introduces a dual-threshold mechanism for the generation of BBAs within the Openmax framework, aiming to advance OSR capabilities across various deep neural networks. Here, we provide a succinct overview of evidence theory:

A. Basis of Evidence Theory

DST provides a robust theoretical framework for modeling and reasoning under uncertainty. In DST, the Frame Of Discernment (FOD) Θ contains l mutually exclusive and exhaustive elements $\Theta = \{\theta_1, \theta_2, \ldots, \theta_l\}$. The power set of Θ (the set of all subsets of Θ) is denoted by 2^{Θ}. The BBA, also known as a mass function, m, maps from 2^{Θ} to the interval $[0, 1]$, adhering to the conditions:

$$\sum\nolimits_{A \subseteq \Theta} m(A) = 1, \quad m(\emptyset) = 0 \tag{11}$$

here, $m(A)$ quantifies the degree of belief supporting the proposition A. A subset A is designated as a focal element if $m(A) > 0$.

Dempster's rule of combination, a mechanism for combining two distinct sources within the DST framework, is articulated as

$$m_1 \oplus m_2(A) = \begin{cases} 0, A = \emptyset \\ \frac{1}{1-K} \sum\limits_{B \cap C = A} m_1(B) m_2(C), A \neq \emptyset \end{cases} \tag{12}$$

where $K = \sum_{B \cap C = \emptyset} m_1(B) m_2(C)$ signifies the aggregate conflict or contradiction between mass assignments.

For probabilistic decision-making based on BBA, Smets et al. introduced the pignistic probability transformation to convert a BBA into a probability measure $BetP$ [19]:

$$BetP(\theta_i) \triangleq \sum\nolimits_{\theta_i \in A} \frac{m(A)}{|A|}, \forall \theta_i \in \Theta \tag{13}$$

where $|A|$ represents the cardinality of A. Decisions are typically rendered by selecting the hypothesis in FOD with the highest $BetP$ value.

B. Generation of BBA Based on Dual Thresholds

Algorithm 1 Generation of BBA for the Openmax networks

Input: j-th Openmax network output highest posterior probability $\hat{P}_j(y = y^*|x)$ and its corresponding class y_j^* for sample x
$j = 1, 2, ..., N_d$, N_d refers to the number of different DNNs used;

Output: BBAs from the Openmax networks $m_j(\theta_1), m_j(\theta_2)$ and $m_j(\theta_1, \theta_2)$
$j = 1, 2, ..., N_d$, corresponding to the j-th Openmax network;

1: **for** $j = 1, 2, \cdots, N_d$ **do**
2: **if** $y_j^* = N + 1$ **then**
3: Following the Openmax's criteria in Eq.(9), treat the sample as the unknowns:

$$m_j(\theta_1) = 0, m_j(\theta_2) = 1; \tag{14}$$

4: **else**
5: Assign values to $m_j(\theta_1), m_j(\theta_2)$ based on the dual-threshold ϵ_1, ϵ_2:

$$\begin{aligned} m_j(\theta_1) &= \frac{1}{1 + e^{-\beta \times \left(\hat{P}_j(y = y^*|x) - \epsilon_1\right)}} \\ m_j(\theta_2) &= \frac{1}{1 + e^{\ \beta \times \left(\hat{P}_j(y = y^*|x) - \epsilon_2\right)}} \end{aligned} \tag{15}$$

where, β acts as a slope hyperparameter;
6: **end if**
7: Assign values to $m_j(\theta_1, \theta_2)$:

$$m_j(\theta_1, \theta_2) = 1 - m_j(\theta_1) - m_j(\theta_2) \tag{16}$$

8: **end for**
9: **return** $m_j(\theta_1), m_j(\theta_2)$ and $m_j(\theta_1, \theta_2), j = 1, 2, \cdots, N_d$.

Our methodological refinement aims to use the evidence theory for an appropriate representation of the uncertainties near decision thresholds. By leveraging compound focal elements from evidence theory, we delineate the indistinct boundary separating known and unknown classes. In our FOD, θ_1 represents samples classified within the known class, while θ_2 is designated for samples considered as belonging to the unknown class. The compound focal element $\{\theta_1, \theta_2\}$ represents the degree of indistinguishability between internal known classes and external unknowns.

In the training phase, consistent with the original Openmax framework, we train multiple deep neural networks with a Softmax layer. For each DNN, we calculate MAVs and employ the Weibull distribution, thus equipping the network with Openmax layer for open set recognition. The aforementioned process is illustrated in Part A of Fig. 1. Different DNNs with Openmax layers, also called as Openmax networks, are capable of capturing activation features from diverse approaches.

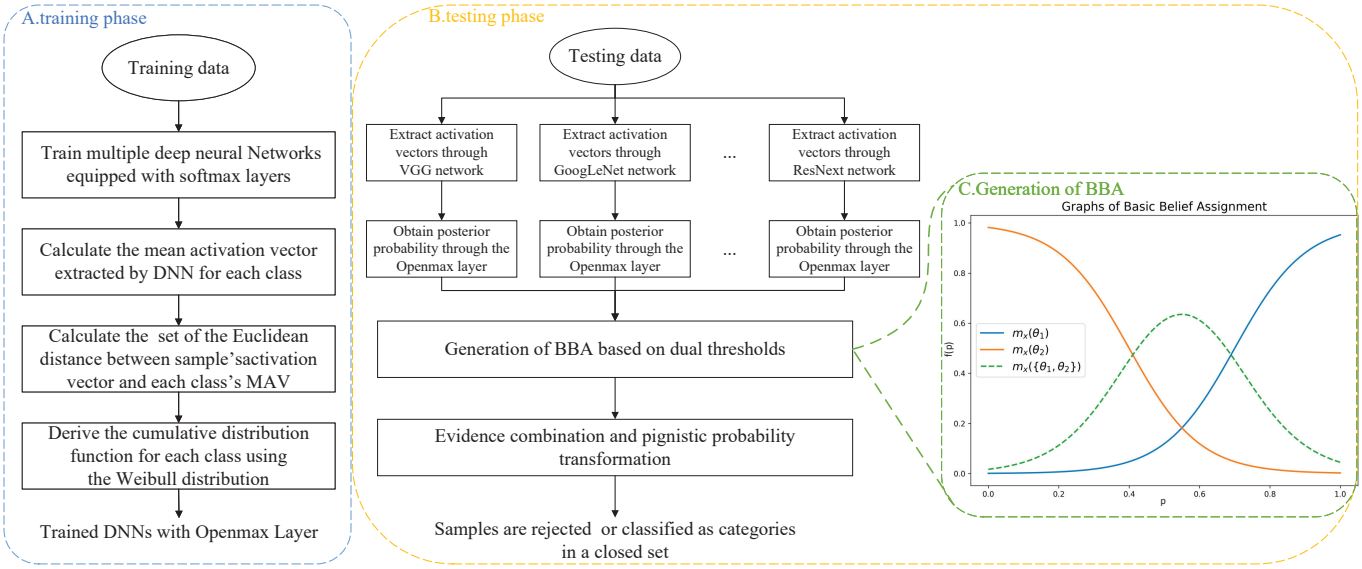

Fig. 1. Block Diagram of the Dual-Threshold Openmax with Evidence.

Subsequently, we will fuse them to boost the performance of OSR.

In the testing phase, Each Openmax network individually processes a new test sample x, deriving the corresponding activation vector and network output probabilities. Based on Openmax network outputs, our overall BBAs generation algorithm is summarized in Algorithm 1. Here are some details from our algorithm:

For Eq.(14), when $y^* = N + 1$, our approach remains consistent with the original Openmax methodology, treating the sample as the unknowns with rejection from Eq.(9). Our efforts within the Openmax framework focus on the threshold-based rejection criteria, detailed in Eq.(10).

In Eq.(15) and (16), our method employs a dual-threshold approach, incorporating both high and low thresholds ($\epsilon_1, \epsilon_2; \epsilon_1 > \epsilon_2$). The choice of dual thresholds allows for increased flexibility. In this case, a straightforward and intuitive approach is employed to determine the upper and lower thresholds by adjusting the original threshold; this adjustment involves adding or subtracting 0.1 ($\epsilon_1, \epsilon_2 = \epsilon \pm 0.1$). Here, the slope hyperparameter β is usually set to 10, ensuring that values of $m_j(\theta_1)$ and $m_j(\theta_2)$ remain within a reasonable range in (0,1) when $\hat{P}(y = y^*|x)$, abbreviated as $\hat{P}$, varies between [0,1]. To generate a legal BBA, by differentiating function $m_j(\theta_1)+m_j(\theta_2)$ with respect to $\hat{P}$, its maximum value is less than 1. While, $m_j(\theta_1, \theta_2)$ consistently stays above 0. In part C of Fig. 1, the membership curve for the generation of BBA draws inspiration from the reference [20]. This framework introduces flexibility beyond the original Openmax's single threshold, which rigidly separates classes.

When $\hat{P}$ matches ϵ_1, the value for $m_j(\theta_1)$ is 0.5. As the value of $\hat{P}$ surpasses this threshold, $m_j(\theta_1)$ increases above 0.5, progressively nearing 1, considering the sample increasingly likely to be known. When $\hat{P}$ equals ϵ_2, $m_j(\theta_2)$ is also 0.5. Below the threshold, $m_j(\theta_2)$ climbs above 0.5 and steadily approaches 1, suggesting the sample increasingly likely to be unknown. The dual-threshold model naturally creates a decision buffer zone between its two thresholds. This zone provides additional leeway and allows for a "waiting"

Algorithm 2 OSR Based on Evidence Combination

Input: Alg.1 output BBAs $m_j(\theta_1), m_j(\theta_2)$ and $m_j(\theta_1, \theta_2)$; j-th DNN output activation vectors $AV_j(x)$; $j = 1, 2, ..., N_d$, N_d refers to the number of different DNNs used;

Output: Reject input or the input sample class $y^{*'}$;

1: Combine BBAs via Dempster's rule in Eq.(12):

$$m(A) = m_1 \oplus m_2 \oplus ...m_j(A) \tag{17}$$

2: Convert $m(A)$ into a probability measure $BetP(\theta_i)$ using Smets' method in Eq.(13)

3: **if** $BetP(\theta_2) > BetP(\theta_1)$ **then**

4: **return** Reject input

5: **else**

6: Average the activation vectors from different DNNs:

$$MAV(x) = \frac{1}{N_d} \sum_{i=1}^{N_d} AV_i(x) \tag{18}$$

7: Classify the sample x into a specific known class $y^{*'}$:

$$P(y = j|x) = \frac{e^{mav_j(x)}}{\sum_{j=1}^{N} e^{mav_j(x)}}$$
$$y^{*'} = argmax_j P(y = j|x) \tag{19}$$

8: **return** $y^{*'}$.

9: **end if**

period for more information from various Openmax networks to arrive.

Moving forward, we conduct OSR using a decision fusion method, which is summarized in Algorithm 2. Regarding Eq.(17) and subsequent details, Dempster's combination rule and Smets' probabilistic transformation are established and reliable techniques in decision fusion. Employing these methods, we decide whether to reject sample x or recognize it as belonging to a known class. For Eq.(18) and (19), when the sample is not rejected, the aggregate activation vectors inform the final classification decision in the Softmax function.

Part B of Fig. 1 illustrates the aforementioned process in Alg. 1 and 2. In the complex and ever-changing real-world scenarios, this fusion approach prevents the frequent decision flips that might occur with the original Openmax in the face of slight information variances. By integrating a decision buffer zone and leveraging evidence theory for multi-model fusion, the method in the paper substantially improves the robustness and reliability of open set recognition, which is demonstrated in the experimental section.

IV. Experiments for Open Set Recognition

A. Evaluation Metrics

Before discussing evaluation metrics, we reference the concept of Openness (O, [2]) in open set recognition to quantify the degree of challenge posed by the openness of the recognition scenario:

$$O = 1 - \sqrt{\frac{2 \times N_{\text{train}}}{N_{\text{test}} + N_{\text{target}}}} \tag{20}$$

here, N_{train} represents the number of training classes seen during training, N_{test} denotes the number of unknown test classes that will be observed during testing, and N_{target} refers to the number of target classes that need to be correctly recognized during testing. In the CIFAR+10 protocol discussed in this paper, four classes were used as the closed set for training, while ten classes were treated as unknown classes. Here, $N_{\text{target}} = 14$, $N_{\text{train}} = 4$, and $N_{\text{test}} = 4$, resulting in an Openness of 33.33%.

In our methodology, we introduce ACCuracy (ACC) as a fundamental evaluation metric, delineating the peak classification probability achievable across the spectrum of potential thresholds ϵ. This metric is particularly tailored for scenarios involving open set recognition with N predefined classes in the closed set, and can be mathematically represented as:

$$\text{ACC} = \frac{\sum_{i=1}^{\mathcal{D}_{\mathcal{T}}} \mathbf{1}(\hat{y}_i = k | x \in \mathcal{D}_{\mathcal{T}}^k) + \sum_{i=1}^{\mathcal{D}_U} \mathbf{1}(\hat{y}_i = N+1) | x \in \mathcal{D}_U)}{\left|\mathcal{D}_{\mathcal{T}}^k\right| + |\mathcal{D}_U|} \tag{21}$$

here, $\mathcal{D}_{\mathcal{T}}^k$ and $\mathcal{D}_U$ respectively signify the sets of samples belonging to the k-th closed set class and those from the open set. Accuracy thus provides an intuitive gauge of an algorithm's efficacy, though its reliability hinges on the selection of an appropriate threshold.

Considering the inherent unpredictability of unknown sample prevalence in real-world applications, the utilization of arbitrary thresholds for OSR evaluations becomes untenable. Consequently, we incorporate the Area Under the Receiver Operating Characteristic (AUROC) curve as a critical evaluation metric. The AUROC, independent of any threshold setting, contrasts the true positive rate against the false positive rate by varying the threshold. This metric is interpreted as the probability of a positive instance receiving a higher score than a negative instance, offering a robust measure of an algorithm's ability to distinguish between known and unknown classes.

However, while the AUROC adeptly distinguishes between known and unknown classes, it overlooks the accuracy of classifying known classes in OSR scenarios— a critical aspect often obscured by this widely accepted metric. To address this, we introduce the Open Set Classification Rate (OSCR [21]) as an innovative evaluation metric. OSCR, too, measures the area under a curve drawn with the Correct Classification Rate (CCR) on the y-axis against the False Positive Rate (FPR) on the x-axis. CCR is defined as:

$$CCR(\delta) = \frac{\left| x \in \mathcal{D}_{\mathcal{T}}^k \wedge \underset{k}{\text{argmax}} P(k|x) = \hat{k} \wedge P(\hat{k}|x) \geq \delta \right|}{\left|\mathcal{D}_{\mathcal{T}}^k\right|} \tag{22}$$

where δ is the score threshold. The FPR is similarly defined as:

$$FPR(\delta) = \frac{|x | x \in \mathcal{D}_U \wedge \max_k P(k|x) \geq \delta|}{|\mathcal{D}_U|} \tag{23}$$

A higher OSCR value signifies superior performance in open set recognition, capturing the appropriate balance between correctly classifying known classes and effectively identifying unknowns.

B. Dataset

Our experimental evaluation leverages the CIFAR-10 and CIFAR-100 datasets [22], renowned benchmarks in image classification that are instrumental in assessing the performance of various algorithms and pose significant challenges. Adhering to the methodologies outlined in prior research [23], we implement two distinct protocols for our assessments:

CIFAR-10: Using four classes from the CIFAR-10 dataset as a closed set for training purposes, we conduct open set recognition across all ten classes, culminating in an openness of 33.33%.

CIFAR+10, CIFAR+50: In the CIFAR+M setup, we select four classes from CIFAR-10 for training and designate M non-overlapping classes from CIFAR-100 as unknowns to simulate open set conditions. The calculated openness for the CIFAR+10 and CIFAR+50 configurations are 33.33% and 62.86%, respectively, showcasing varying degrees of challenge and complexity in open set recognition scenarios.

C. Experimental Results

We trained the five deep neural networks mentioned in Section II-B, employing stochastic gradient descent [24] as the optimizer across all networks. Our experiments were

TABLE I
THE ACC, AUROC AND OSCR CURVE RESULTS OF OPEN SET RECOGNITION

Method	CIFAR10			CIFAR+10			CIFAR+50		
	ACC(%)	AUROC	OSCR	ACC(%)	AUROC	OSCR	ACC(%)	AUROC	OSCR
Original Openmax	66.62± 3.56	68.79	84.02	72.21± 3.07	80.72	89.78	70.65± 3.15	79.94	88.73
Weighted Fusion Openmax	70.71± 3.08	75.27	85.24	75.99± 2.62	86.16	91.12	74.28± 2.68	84.48	90.26
DST-enhanced Openmax	**72.23 ± 2.97**	**78.80**	**85.68**	**77.35± 2.64**	**87.28**	**92.35**	**75.63± 2.61**	**85.12**	**91.67**

repeated randomly five times, with the variance of the accuracy metric systematically calculated. Our experimental findings are presented in Table 1. In this table, the "Original Openmax" method refers to the conventional Openmax approach that utilizes ResNet for feature extraction. The "DST-enhanced Openmax" represents the improved methodology proposed in this study. Beyond our DST-enhanced Openmax method, we also explored a simple weighted fusion approach to combine outputs from different Openmax networks. This method averages the outputs of various Openmax networks for fusion, as expressed mathematically by:

$$M\hat{P}(y=j|x) = \frac{1}{N_d}\sum_{i=1}^{N_d}\hat{P}_i(y=j|x) \tag{24}$$

here, N_d also refers to the number of different DNNs used. In this experiment, $N_d = 5$. This approach, labeled as "Weighted Fusion Openmax" in Table 1, leverages information from diverse deep networks to enhance decision-making.

As illustrated in Table 1, our method, which enhances the Openmax approach with a dual-threshold mechanism based on evidence theory, outperforms the original and simple fusion methods across all three metrics in three experiments. Our approach not only achieves higher intuitive accuracy but also excels in threshold-independent metrics such as AUROC and OSCR. This demonstrates the robustness of our method and its capability to effectively balance the classification of closed-set classes with the rejection of open-set classes, underscoring its superiority in managing the challenges of open set recognition.

V. CONCLUSION

In this paper, our approach refines the Openmax framework with a dual-threshold mechanism based on DST, adept at managing uncertainty. This novel strategy surpasses the original model's single-threshold reliance, accurately demarcating the fuzzy boundary between known and unknown classes. By combining BBAs from multiple deep neural networks, we have effectively improved our open set recognition capabilities. In future, we intend to refine and compare our enhanced Openmax approach by examining a wider range of combination rules, including PCR6 [25], to further improve OSR performance. Additionally, we will enhance Openmax by integrating generalized evidence theory, which permits the allocation of generalized BBAs to the empty subset, quantifying support for unknown class.

REFERENCES

[1] D. Hendrycks and K. Gimpel, "A baseline for detecting misclassified and out-of-distribution examples in neural networks," *arXiv preprint arXiv:1610.02136*, 2016.

[2] W. J. Scheirer, A. de Rezende Rocha, A. Sapkota, and T. E. Boult, "Toward open set recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 7, pp. 1757–1772, 2012.

[3] L. M. Manevitz and M. Yousef, "One-class svms for document classification," *Journal of machine Learning research*, vol. 2, no. Dec, pp. 139–154, 2001.

[4] S. Kotz and S. Nadarajah, *Extreme value distributions: theory and applications*. world scientific, 2000.

[5] W. J. Scheirer, L. P. Jain, and T. E. Boult, "Probability models for open set recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 11, pp. 2317–2324, 2014.

[6] F. Li and H. Wechsler, "Open set face recognition using transduction," *IEEE transactions on pattern analysis and machine intelligence*, vol. 27, no. 11, pp. 1686–1697, 2005.

[7] H. Zhang and V. M. Patel, "Sparse representation-based open set recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 8, pp. 1690–1696, 2016.

[8] A. Bendale and T. E. Boult, "Towards open set deep networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1563–1572, 2016.

[9] Z. Ge, S. Demyanov, Z. Chen, and R. Garnavi, "Generative openmax for multi-class open set classification," *arXiv preprint arXiv:1707.07418*, 2017.

[10] A. H. Oveis, E. Giusti, S. Ghio, and M. Martorella, "Extended openmax approach for the classification of radar images with a rejection option," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 59, no. 1, pp. 196–208, 2022.

[11] J. Shao, Z. Song, J. Wu, and W. Shen, "Openfe: feature-extended openmax for open set facial expression recognition," *Signal, Image and Video Processing*, pp. 1–10, 2023.

[12] G. Shafer, *A mathematical theory of evidence*, vol. 42. Princeton university press, 1976.

[13] W. J. Scheirer, A. Rocha, R. J. Micheals, and T. E. Boult, "Meta-recognition: The theory and practice of recognition score analysis," *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 8, pp. 1689–1695, 2011.

[14] A. Sengupta, Y. Ye, R. Wang, C. Liu, and K. Roy, "Going deeper in spiking neural networks: Vgg and residual architectures," *Frontiers in neuroscience*, vol. 13, p. 95, 2019.

[15] P. Ballester and R. Araujo, "On the performance of googlenet and alexnet applied to sketches," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 30, 2016.

[16] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

[17] G. Huang, Z. Liu, G. Pleiss, L. Van Der Maaten, and K. Q. Weinberger, "Convolutional networks with dense connectivity," *IEEE transactions on pattern analysis and machine intelligence*, vol. 44, no. 12, pp. 8704–8716, 2019.

[18] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1492–1500, 2017.

[19] P. Smets and R. Kennes, "The transferable belief model," *Classic Works of the Dempster-Shafer Theory of Belief Functions*, pp. 693–736, 2008.

[20] J. Dezert, Z.-g. Liu, and G. Mercier, "Edge detection in color images based on dsmt," in *14th International Conference on Information Fusion*, pp. 1–8, IEEE, 2011.
[21] A. R. Dhamija, M. Günther, and T. Boult, "Reducing network agnostophobia," *Advances in Neural Information Processing Systems*, vol. 31, 2018.
[22] A. Krizhevsky, G. Hinton, *et al.*, "Learning multiple layers of features from tiny images," 2009.
[23] P. Oza and V. M. Patel, "C2ae: Class conditioned auto-encoder for open-set recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2307–2316, 2019.
[24] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proceedings of COMPSTAT'2010: 19th International Conference on Computational StatisticsParis France, August 22-27, 2010 Keynote, Invited and Contributed Papers*, pp. 177–186, Springer, 2010.
[25] F. Smarandache and J. Dezert, "On the consistency of pcr6 with the averaging rule and its application to probability estimation," in *Proceedings of the 16th International Conference on Information Fusion*, pp. 1119–1126, IEEE, 2013.