Pattern Recognition 146 (2024) 110046

Contents lists available at ScienceDirect

Pattern Recognition

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

Conditional feature generation for transductive open-set recognition via dual-space consistent sampling

Jiayin Sun [a,b], Qiulei Dong [a,b,c,*]

[a] National Laboratory of Pattern Recognition, CASIA, Beijing, 100190, China
[b] School of Artificial Intelligence, UCAS, Beijing, 100049, China
[c] Center for Excellence in Brain Science and Intelligence Technology, CAS, Beijing, 100190, China

ARTICLE INFO

Dataset link: https://github.com/sjy-1995/IT-OSR-code

Keywords:
Open-set recognition
Transductive learning
Generative learning

ABSTRACT

Open-set recognition (OSR) aims to simultaneously detect unknown-class samples and classify known-class samples. Most of the existing OSR methods are inductive methods, which generally suffer from the domain shift problem that the learned model from the known-class domain might be unsuitable for the unknown-class domain. Addressing this problem, inspired by the success of transductive learning for alleviating the domain shift problem in many other visual tasks, we propose an Iterative Transductive OSR framework, called IT-OSR, which implements three explored modules iteratively, including a reliability sampling module, a feature generation module, and a baseline update module. Specifically at the initialization stage, a baseline method, which could be an arbitrary inductive OSR method, is used for assigning pseudo labels to the test samples. At the iteration stage, based on the consistency of the assigned pseudo labels between the output/logit space and the latent feature space of the baseline method, a dual-space consistent sampling approach is presented in the reliability sampling module for sampling some reliable ones from the test samples. Then in the feature generation module, a conditional dual-adversarial generative network is designed to generate discriminative features of both known and unknown classes. This generative network employs two discriminators for implementing fake/real and known/unknown-class discriminations respectively. And it is trained by jointly utilizing the test samples with their pseudo labels selected in the reliability sampling module and the labeled training samples. Finally in the baseline update module, the above baseline method is updated/re-trained for sample re-prediction by jointly utilizing the generated features, the selected test samples with pseudo labels, and the training samples. Extensive experimental results on both the standard-dataset and the cross-dataset settings demonstrate that the derived transductive methods, by introducing two typical inductive OSR methods into the proposed IT-OSR framework, achieve better performances than 19 state-of-the-art methods in most cases.

1. Introduction

In many real-world scenarios, the deployment of image recognition models is required under open-set conditions. This encourages more and more researchers in the fields of pattern recognition and computer vision to investigate the open-set recognition (OSR) problem, which aims to correctly identify unknown-class samples and maintain the classification accuracy for known-class samples [1–23].

Existing OSR methods can be roughly divided into two groups: the discriminative OSR methods [1–10,16] which directly learn the classifiers for recognizing different classes based on the discriminative representations, and the generative OSR methods [11–13,15,17–19] which model the distributions of the known-class samples. However, most of these methods [1–9,11–13,15–19] employ the inductive OSR strategy, which uses only the labeled known-class training samples for model learning. They have to suffer from the domain shift problem, where their learned recognition models from the known-class domain are not suitable for the unknown-class domain.

Different from the inductive learning strategy, the transductive learning strategy makes use of both training and test samples for model learning. Recently, transductive learning has demonstrated its effectiveness for alleviating the domain shift problem in various visual tasks [24–26]. But to our knowledge, only a pioneering work [10] investigated transductive OSR, where a model (called S2OSC) was trained in a transductive manner. It was implemented in the two following steps: Firstly, it provided pseudo labels for the test samples that were predicted as unknown classes with high confidence scores by

* Correspondence to: National Laboratory of Pattern Recognition, CASIA, China.
E-mail addresses: jiayin.sun@nlpr.ia.ac.cn (J. Sun), qldong@nlpr.ia.ac.cn (Q. Dong).

https://doi.org/10.1016/j.patcog.2023.110046
Received 26 August 2022; Received in revised form 30 August 2023; Accepted 12 October 2023
Available online 16 October 2023

a baseline method; Then, the baseline model was re-trained by jointly utilizing the pseudo-labeled test samples, the remaining unlabeled test samples, and the labeled known-class samples from the original training set. However, the following two open problems still remain for the transductive OSR task:

Q1: The pseudo-labeled data selection problem: In general, some test samples are inevitably misclassified during the transductive process at the training stage, resulting in a poor classification model by utilizing them for training. This issue raises the problem: "How to select a relatively reliable subset from the whole set of the pseudo-labeled test samples for model training?"

Q2: The sample imbalance problem: The number of the pseudo-labeled unknown-class test samples at the training stage is generally much smaller than that of the known-class training samples, so that it is prone to train a poor classification model by jointly utilizing these imbalanced samples. This issue raises the problem: "How to automatically obtain a larger and balanced number of data from the given imbalanced samples for model training?"

Addressing the aforementioned domain shift problem as well as Problems Q1 and Q2, we propose an Iterative Transductive OSR framework, called IT-OSR, consisting of a reliability sampling module, a feature generation module, and a baseline update module. Under the proposed IT-OSR framework, the three modules are implemented in an iterative manner. At the initialization stage, we introduce a baseline inductive OSR method for assigning pseudo labels to the test samples (in fact, an arbitrary inductive OSR method in literature could be also used as the baseline method here). At the iterative training stage, addressing Problem Q1, a dual-space consistent sampling approach is firstly explored to select a relatively more reliable subset of samples from the test dataset in the reliability sampling module. It is noted that the traditional sampling strategy generally identifies samples with high logit values as reliable ones. Unlike the traditional sampling strategy, the dual-space consistent sampling approach identifies reliable samples whose pseudo labels assigned in the output space are consistent with those of most of their neighbors in the latent feature space of the baseline methods. Then addressing Problem Q2, a conditional dual-adversarial generative network is designed to synthesize features of both known and unknown classes in the feature generation module, which is trained on both the training and selected test samples. Unlike conventional adversarial generative networks, the designed network contains a feature generator but two discriminators, one for discriminating fake features from real ones while the other for discriminating known-class features from unknown-class ones. Finally, the baseline method is updated and the labels of the test samples are re-predicted for the next iteration in the baseline update module.

In sum, the main contributions of this paper are three-fold:

- We explore the dual-space consistent sampling approach for handling the aforementioned Problem Q1. Since this approach makes use of the consistency of the assigned pseudo labels between two different spaces as described above, it could improve the sampling performance in comparison to traditional sampling strategy as demonstrated by the experimental results in Section 4.7.
- We design the conditional dual-adversarial generative network for feature generation. Due to its special architecture that contains a feature generator, a fake/real discriminator, and a known/unknown-class discriminator, it could not only enlarge the number of both known-class and unknown-class features for alleviating the aforementioned Problem Q2, but also increase the variety of unknown-class features, as demonstrated by the results in Section 4.7.
- We propose the IT-OSR framework, based on the explored dual-space consistent sampling approach and the designed conditional dual-adversarial generative network. The proposed IT-OSR framework could accommodate an arbitrary inductive OSR method as its baseline method, and its effectiveness is demonstrated by the experimental results in Section 4.

The remainder of this paper is organized as follows. Some existing inductive OSR methods are reviewed in Section 2. The proposed framework is described in detail in Section 3. Experimental results are reported in Section 4. Section 5 concludes the paper.

2. Related works

In this section, we firstly review existing OSR methods, then we review some open-set approaches that train on synthetic data in other tasks.

2.1. OSR methods

As discussed above, to our knowledge, only a pioneering transductive OSR method [10] has been proposed in literature. Since a transductive method generally needs to use an inductive method as its baseline method, here, we give a detailed review on the existing works for inductive OSR from the following two aspects.

Discriminative Inductive OSR Methods. The discriminative methods directly learn classification networks. They aim to boost the discriminability of the network representations in training, such that the known-class samples or the unknown-class samples can be classified or detected correctly based on the learnt network representations in inference. Bendale and Boult [1] proposed OpenMax, where an OpenMax layer that calculated open-set probabilities based on the distances to each known-class center was designed. They replaced the SoftMax layer in the traditional closed-set classification network with their designed OpenMax layer, aiming to expand the closed-set calculation of SoftMax to open-set distance-based calculation. Jang and Kim [16] replaced the SoftMax layer with a set of one-vs-rest networks (OVRNs), the decisions from which are combined for alleviating the overgeneralization of the SoftMax layer which would cause high confidence scores for unknown-class test samples. Miller et al. [2] proposed class anchor clustering loss, which constrained the known-class representations in the logit space. Zhou et al. [3] proposed to reserve placeholders for unknown classes during model training by adding an additional output and mix upping features. Yoshihashi et al. [4] proposed a joint training framework where classification and reconstruction were simultaneously implemented, and the model prediction vector concatenated with latent features was used for classification. Oza and Patel [5] proposed a class conditioned autoencoder, which detected unknown classes by training with reconstruction errors of the mismatched-class image pairs. Perera et al. [6] proposed to add the reconstructed images of the known classes as additional channels of the 3-channel input images for model training. Chen et al. [7] proposed to learn discriminative reciprocal points by digging 1-vs-rest information among the known classes for reserving the extra-class space as the open space, and they also proposed an adversarial version called A-RPL in [8]. Perera and Patel [9] proposed to train images ensembled with different transformed versions by extreme geometric transformations for mining more information from the original known-class images.

Generative Inductive OSR Methods. The generative methods utilize either GANs [27] or other generative models to model the distributions of the known classes. Some methods modeled the known-class distributions either explicitly or implicitly, and then detected the unknown classes by their inconsistency with these distributions. Sun et al. [11] proposed to model each known class to be subject to a Gaussian, thus the representations of the test samples deviating from the modeled Gaussians could be regarded as unknown classes. Similarly, Guo et al. [15] proposed to use a capsule network to better model each known class as a Gaussian. Besides, the works in [17,18] used a Gaussian mixture distribution to model the feature distribution of each known class, and Sun et al. [19] proposed to use multiple mixtures of exponential power distributions to model the known-class feature distributions. Zhang et al. [12] proposed to apply a flow-based model for hybrid training of both classification and generation. This model could output the probability of the test sample belonging to the known classes. Kong and Ramanan [14] trained a GAN based on the adversarial training between the known-class images/features and the selected outlier images/features. The discriminator of the trained GAN was used for detecting unknown classes. Besides, some methods further generated unknown-class samples based on the modeled known-class distributions, and then trained open-set classifiers using these generated unknown-class samples as an additional class. Neal et al. [13] proposed to generate counterfactual images as unknown classes which were easily confused with the known classes in the feature space. Chen et al. [8] also proposed an enhanced version from A-RPL, called ARPL-CS, which generated diverse and confusing samples for data augmentation.

2.2. Open-set approaches that train on synthetic data in other tasks

Recently, there are some open-set approaches that also train on synthetic data in other tasks, *e.g.*, out-of-distribution detection. Here, we briefly review these methods.

Lee et al. [28] proposed to generate 'boundary' samples in the low-density of the training distribution by training a GAN with a confidence loss, and jointly trained a classifier to be less confident for the generated samples. Grcić et al. [29] trained a normalized flow network by constraining the distribution of the generated data to resemble that of the training data. Simultaneously, they jointly trained a classifier, which constrained the generated data to be classified with high entropy. The two opposed losses could generate samples at the boarder of the training distribution. Zhao et al. [30] proposed to finetune the discriminator by an implicit generator, where the implicit generator was trained to generate samples with low predictive entropy, while the discriminator was trained to maximizing the predictive entropy of each generated sample. Du et al. [31] modeled the in-distribution (ID) features as class-conditional Gaussians and sampled virtual outliers from the low-likelihood region. Then, they designed an unknown-aware training loss for both producing a low out-of-distribution (OOD) score on ID data and producing a high OOD score on virtual outliers.

It is noted from the above description that all the above methods were explored under the inductive setup, and their main difference is that they use different feature generation manners to generate synthetic data. In comparison to them, our main contributions are two-fold:

(i) We propose a new feature generation module (where a conditional dual-adversarial generative network is designed) under the transductive setup. This module is completely different from the used feature generation manners in [28–31].

(ii) Considering that some of the predicted pseudo labels during the iterative process of transductive learning are usually unreliable, we propose a reliability sampling module for selecting a more reliable test samples, which are more helpful for training the above feature generation module. There is no such a module with a similar function in [28–31].

3. Methodology

In this section, we propose the IT-OSR framework that iteratively implements three explored modules: a reliability sampling module, a feature generation module, and a baseline update module. Firstly, the pipeline of the proposed IT-OSR is introduced. Then, the three explored modules are described respectively in detail. Finally, two novel transductive OSR methods are derived from the proposed IT-OSR framework.

Algorithm 1 The IT-OSR framework

```
Input: The labeled training set $\mathbf{D}^l$, the unlabeled test dataset $\mathbf{D}^u$, and the initial
    baseline model $M_0$
Output: The predictions $P_T^u$ on the test dataset $\mathbf{D}^u$ made by the updated model
    at the $T$-th iteration
1: Initialization: Make predictions on $\mathbf{D}^u$ by $M_0$;
2: for $t = 1$ to $T$ do
3:     Reliability Sampling Module: Select a subset $\mathbf{D}_{s_t}^p$ of test samples from $\mathbf{D}^u$
       based on the dual-space consistent sampling approach;
4:     Feature Generation Module: Train the conditional dual-adversarial generative network under a one-hot condition with $\mathbf{D}^l \cup \mathbf{D}_{s_t}^p$ and generate the
       generated set $\mathbf{D}_{g_t}$ of features from the generator;
5:     Baseline Update Module: Obtain the updated baseline model $M_t$ by re-training the baseline model with $\mathbf{D}^l \cup \mathbf{D}_{s_t}^p \cup \mathbf{D}_{g_t}$ and make predictions $P_t^u$ on
       $\mathbf{D}^u$ by $M_t$;
6: end for
7: return $P_T^u$;
```

3.1. Pipeline of IT-OSR

The pipeline of the proposed IT-OSR framework which consists of a reliability sampling module, a feature generation module, and a baseline update module, is shown in Fig. 1. As seen from this figure, at the initialization stage, there is a baseline method M_0 that consists of a feature extractor F and a classifier C. The features are extracted from input images via the feature extractor F, and are fed into the classifier C for prediction.

Then, the three explored modules are applied in an iterative manner: At the tth ($t = 1, 2..., T$, and T represents a preset maximum iteration number) iteration, firstly in the reliability sampling module, a relatively reliable subset $\mathbf{D}_{s_t}^p$ of the test samples is selected from the unlabeled test dataset $\mathbf{D}^u$ by an explored dual-space consistent sampling approach. Then, in the feature generation module, we design a conditional dual-adversarial generative network to generate both known-class and unknown-class features. This generative network is trained with both the training set $\mathbf{D}^l$ and the sampled test set $\mathbf{D}_{s_t}^p$ with pseudo labels. After training the generative network, we generate both known-class and unknown-class features, the synthesized set is denoted as $\mathbf{D}_{g_t}$. Finally in the baseline update module, the baseline model is updated (re-trained) according to the union set $\mathbf{D}^l \cup \mathbf{D}_{s_t}^p \cup \mathbf{D}_{g_t}$ for making predictions P_t^u on the test dataset $\mathbf{D}^u$. The iterative process would not be terminated until the iteration number t reaches a preset maximum T, and the assigned labels to all the test samples by the re-trained baseline method at the final iteration are used as the final predictions. The whole above process of IT-OSR is also outlined in Algorithm 1. In the following parts, the three explored modules at each iteration in Fig. 1 (*i.e.*, the three key Steps 3–5 in Algorithm 1) are introduced respectively.

3.2. Reliability sampling module

In this module, a dual-space consistent sampling approach is explored for selecting a relatively reliable subset from the test dataset, given the predictions of the baseline model at the previous iteration. This module aims to alleviate the referred pseudo-labeled data selection problem in Section 1 to some extent. Here, only such test samples that have consistent predictions between the output space based on confidence scores and the feature space based on feature distances, are considered as relatively 'reliable' samples by the explored dual-space consistent sampling approach according to the following criterion:

Criterion. *A test sample, whose pseudo label assigned by the confidence scores in the output space is consistent with those of more than half of its spatial neighbors in the feature space, is identified as a relatively reliable sample.*

According to this criterion, the dual-space consistent sampling approach is implemented in the following two steps:

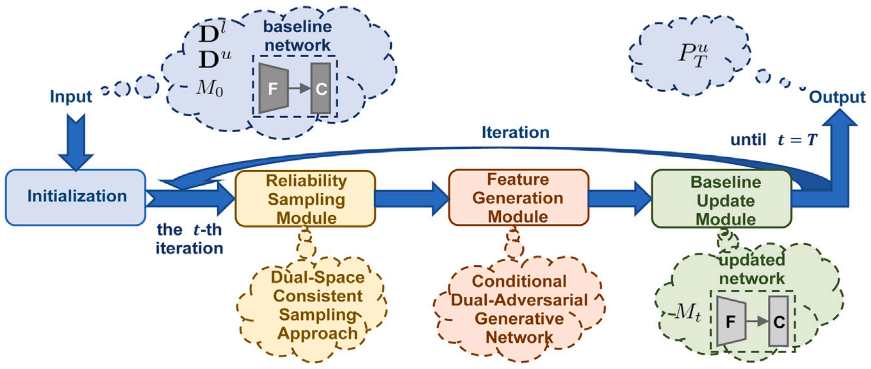


Fig. 1. Pipeline of the proposed IT-OSR framework which implements the reliability sampling module, the feature generation module, and the baseline update module iteratively.

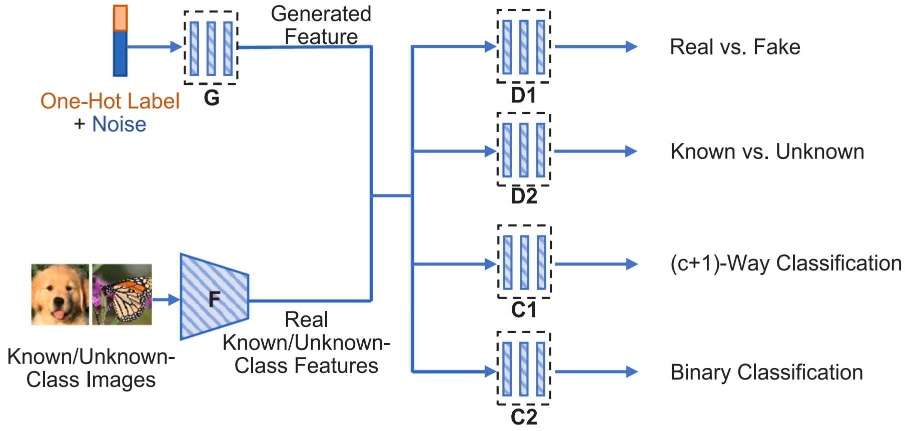


Fig. 2. The architecture of the conditional dual-adversarial generative network.

S1: Given the pseudo labels and confidence scores s calculated in the output space by the updated baseline model at the previous iteration, the test samples are divided into three groups: *known class*, *unknown class*, and *undetermined class*. Such division is based on a predefined threshold, which is statistically calculated from the confidence scores of the training set $\mathbf{D}^l$. The three groups are denoted as $\mathbf{D}^p_k$, $\mathbf{D}^p_{unk}$, and $\mathbf{D}^p_{und}$ respectively. The above grouping process can be formulated as:

$$\begin{cases}\mathbf{D}^p_k, & \text{if } s > \mu + \alpha\delta \\ \mathbf{D}^p_{unk}, & \text{if } s < \mu - \alpha\delta \\ \mathbf{D}^p_{und}, & \text{if } \mu - \alpha\delta \le s \le \mu + \alpha\delta \end{cases} \tag{1}$$

where μ and δ are the mean and standard deviation of the confidence scores calculated from the training set $\mathbf{D}^l$ respectively, and α is a hyper-parameter.

S2: For each sample in $\mathbf{D}^p_k \cup \mathbf{D}^p_{unk}$, we search for its K nearest neighbors from $\mathbf{D}^p_k \cup \mathbf{D}^p_{unk} \cup \mathbf{D}^p_{und}$ in the feature space according to the Euclidean distances. Then, the test samples from $\mathbf{D}^p_k \cup \mathbf{D}^p_{unk}$ whose pseudo labels are consistent with that of more than their $K/2$ neighbors are retained, while those inconsistent with more than their $K/2$ neighbors are removed out of $\mathbf{D}^p_k \cup \mathbf{D}^p_{unk}$. This selected subset of test samples is denoted as $\mathbf{D}^p_{s_t}$.

3.3. Feature generation module

In this module, a conditional dual-adversarial generative network is designed to generate known-class and unknown-class sample features. This network is trained with the features, which are extracted from both the training samples and the selected test samples by the feature extractor F of the updated baseline model. The designed network simply utilizes one-hot vector as the class condition, and introduces this condition to the generative adversarial network in a similar manner to ACGAN [32]. It contains a generator G, a feature extractor F, two discriminators D_1 and D_2, two classifiers C_1 and C_2. The architecture of the conditional dual-adversarial generative network is shown in Fig. 2.

Generator G: As seen from Fig. 2, the generator G is designed to generate known/unknown-class sample features $\mathbf{D}_{g_t}$. The structure of the generator input is a vector, which concatenates a $(c+1)$-dimensional one-hot condition vector and a 128-dimensional Gaussian noise vector. Accordingly, its dimensionality is $c + 1 + 128 = c + 129$. Here, the generator G has a three-layer perceptron architecture. It contains 3 fully-connected layers and uses ReLU (Rectified Linear Unit) as the nonlinear activation function, and the dimensionality of its hidden layers is 4096.

Feature extractor F: The feature extractor F is straightforwardly obtained from the updated baseline model at the previous iteration, whose weights are fixed during the training process of the generative network.

Discriminator $\mathbf{D_1}$: The discriminator D_1 is designed to implement true or false discrimination, whose inputs are the fake features generated by G and the real image features outputted by the feature extractor F, and the dimensionality of its outputs is 1. Here, the discriminator D_1 has the same three-layer perceptron architecture as G except the dimensionalities of the inputs and the outputs.

Discriminator $\mathbf{D_2}$: The discriminator D_2 is designed to implement known-class or unknown-class discrimination for further improving the discriminating ability of the generated features, whose inputs contain known/unknown-class real/generated features, and the dimensionality of its outputs is 1. Here, the discriminator D_2 has the same three-layer perceptron architecture as D_1.

Classifier $\mathbf{C_1}$: The classifier C_1 is designed to classify both the real and the generated features into $(c + 1)$ classes (c known classes and one *unknown class*). Similar to ACGAN [32], this classifier is used to strengthen the feature distinguishability among different known classes and unknown classes. Here, the classifier C_1 has the same three-layer perceptron architecture as D_1 except the dimensionality of the outputs. The dimensionality of its outputs is $(c+1)$.

Classifier $\mathbf{C_2}$: The classifier C_2 is designed to identify whether an input real/generated feature belongs to a known class or not. This is to say, it classifies both the real and generated features into two groups (*known class* and *unknown class*). This classifier is used to pay more attention to the distinguishability of the unknown-class features from the known-class features. Here, the classifier C_2 has the same three-layer perceptron architecture as C_1 except the dimensionality of the outputs. The dimensionality of its outputs is 2.

Similar to WGAN [33], the adversarial game between the generator G and the discriminator D_1 is defined as:

$$\min_G \max_{D_1} V(G, D_1) = \mathbb{E}_{x\sim P_r}\left[D_1(x)\right] - \mathbb{E}_{\tilde{x}\sim P_g}\left[D_1(\tilde{x})\right] \tag{2}$$

where $\mathbb{E}$ is the expectation, P_r and P_g are the data distribution and the generator's distribution respectively, x and $\tilde{x}$ are the extracted real image features and the generated fake features respectively.

Similar to ACGAN [32], the classification loss for the classifier C_1 is defined as:

$$\begin{aligned}\mathcal{L}_{C_1} = &-\mathbb{E}_{x\sim P_r}\left[\log P(y_{c_1} = y_{l_1}|x)\right] \\ &-\mathbb{E}_{\tilde{x}\sim P_g}\left[\log P(y_{c_1} = y_{l_1}|\tilde{x})\right]\end{aligned} \tag{3}$$

where y_{c_1} are the predicted labels predicted by C_1, y_{l_1} are the ground truths for $(c+1)$-way classification.

In addition, the adversarial game between the generator G and the discriminator D_2 is defined as:

$$\begin{aligned}&\min_G \max_{D_2} V(G, D_2) = \\ &-\mathbb{E}_{x\sim P_r}\left[D_2(x|y_{l_1} = c+1) - D_2(x|y_{l_1} \neq c+1)\right] \\ &-\mathbb{E}_{\tilde{x}\sim P_g}\left[D_2(\tilde{x}|\tilde{y} = c+1) - D_2(\tilde{x}|\tilde{y} \neq c+1)\right]\end{aligned} \tag{4}$$

where c is the number of the known classes, $\tilde{y}$ is the class conditions of $\tilde{x}$.

The binary classification loss for the classifier C_2 is defined as:

$$\begin{aligned}\mathcal{L}_{C_2} = &-\mathbb{E}_{x\sim P_r}\left[\log P(y_{c_2} = y_{l_2}|x)\right] \\ &-\mathbb{E}_{\tilde{x}\sim P_g}\left[\log P(y_{c_2} = y_{l_2}|\tilde{x})\right]\end{aligned} \tag{5}$$

where y_{c_2} are the predicted labels predicted by C_2, and y_{l_2} are the ground truths for binary classification. Both y_{l_1} and y_{l_2} are labels (or pseudo labels) of real samples, while y_{l_1} is the $(c+1)$-class one-hot label (the first c classes correspond to c known classes, and the $(c+1)$-th class represents all unknown classes) and y_{l_2} is the 2-class one-hot label (the positive class corresponds to c known classes, and the negative class represents all unknown classes).

Accordingly, the dual-adversarial game is to optimize the objective function $V(G, D_1, D_2)$, which is the weighted sum of the aforementioned two adversarial objective functions $V(G, D_1)$ and $V(G, D_2)$:

$$\begin{aligned}&\min_G \max_{D_1} \max_{D_2} V(G, D_1, D_2) = \\ &\mathbb{E}_{x\sim P_r}\left[D_1(x)\right] - \mathbb{E}_{\tilde{x}\sim P_g}\left[D_1(\tilde{x})\right] + \\ &\lambda\Big\{-\mathbb{E}_{x\sim P_r}\left[D_2(x|y_{l_1} = c+1) - D_2(x|y_{l_1} \neq c+1)\right] \\ &\quad -\mathbb{E}_{\tilde{x}\sim P_g}\left[D_2(\tilde{x}|\tilde{y} = c+1) - D_2(\tilde{x}|\tilde{y} \neq c+1)\right]\Big\}\end{aligned} \tag{6}$$

where λ is a hyper-parameter balancing the weights of the two adversarial games.

Besides, the total classification loss is the sum of the aforementioned two classification losses:

$$\mathcal{L}_{cls} = \mathcal{L}_{C_1} + \mathcal{L}_{C_2} \tag{7}$$

3.4. Baseline update module

In this module, given the training samples, the generated features, and the selected test samples with pseudo labels at the tth iteration, the baseline method is re-trained by utilizing the real/generated unknown-class features as an additional class (the $(c+1)$-th class) for classification. Then, it predicts the pseudo labels P_t^u of all the test samples for the next iteration.

3.5. Two transductive OSR methods derived from IT-OSR

It is noted that the proposed IT-OSR framework could accommodate not only an arbitrary existing inductive OSR method but also a new inductive one, resulting in a transductive OSR method. For evaluating the proposed IT-OSR framework, we explore the following two transductive methods:

(1) **IT-OSR-ARPL**. This transductive method is derived from the IT-OSR framework by simply utilizing the existing inductive method A-RPL [8] as the baseline method.

(2) **IT-OSR-TransP**. We firstly design a new inductive OSR network by simply concatenating the Swin Transformer [34] (used as the feature extractor F) and a three-layer perceptron (used as the classifier C that has the similar architecture to the aforementioned classifier C_1), called TransP. This network is trained with the traditional cross-entropy classification loss. Then, the transductive method IT-OSR-TransP is derived from the proposed IT-OSR framework by utilizing the designed TransP as the baseline method.

4. Experiments

In this section, firstly, we give a brief introduction on the dataset settings and the evaluation metrics as well as the implementation details. Next, we evaluate the proposed IT-OSR framework on coarse-grained datasets under both dataset settings and on fine-grained semantic-shift datasets. Then, we analyze the influence of the maximum iteration number T. Next, we conduct several ablation studies for further evaluating the effect of IT-OSR. Finally, we provide the visualization results of the learned features.

4.1. Dataset settings and evaluation metrics

4.1.1. Dataset settings

To assess open-set recognition performance under different degrees of domain shift, we conduct experiments under two dataset settings. One is the standard-dataset setting, where both the known-class and unknown-class samples are obtained from a same dataset. The other is the cross-dataset setting, where the known-class and unknown-class samples are obtained from two different datasets respectively.

Under the standard-dataset setting, the following six standard datasets are used for evaluation as done in [1–13,15,19]:

- **MNIST** [35]: MNIST is a traditional benchmark dataset for classification containing 10 categories of handwritten digit images, in which 6 categories are randomly chosen as the known classes, while the rest 4 categories as the unknown classes.
- **SVHN** [36]: Similar to MNIST, SVHN also contains 10 categories of digit images but from street view house numbers. It is likewise divided into 6 known classes and 4 unknown classes.
- **CIFAR10** [37]: CIFAR10 is made up of 10 categories of natural images, 6 of which act as the known classes, while the rest 4 act as the unknown classes.
- **CIFAR+10/+50:** CIFAR+10/+50 contains two datasets based on different combinations of CIFAR10 and CIFAR100 [38]. They both randomly select 4 vehicle categories in CIFAR10 as the known classes, and 10/50 random categories in CIFAR100 act as the unknown classes in CIFAR+10/+50.

Table 1
Evaluation on anomaly detection (AUROC) under the standard-dataset setting. The reported results are averaged over the same five trials as [6,9,12,13].

Method	Inductive	Transductive	MNIST	SVHN	CIFAR10	CIFAR+10/+50	TinyImageNet
SoftMax	√	×	0.978	0.886	0.677	0.816/0.805	0.577
OpenMax [1]	√	×	0.981	0.894	0.695	0.817/0.796	0.576
OSRCI [13]	√	×	0.988	0.910	0.699	0.838/0.827	0.586
CROSR [4]	√	×	0.991	0.899	0.883	0.912/0.905	0.589
C2AE [5]	√	×	0.989	0.922	0.895	0.955/0.937	0.748
CGDL [11]	√	×	0.994	0.935	0.903	0.959/0.950	0.762
GCPL [17]	√	×	0.992	0.942	0.883	0.951/0.946	0.759
GMVAE [18]	√	×	0.989	0.941	0.896	0.952/0.947	0.782
MoEP-AE [19]	√	×	0.992	0.965	0.904	0.961/0.962	0.805
GDFR [6]	√	×	–	0.955	0.831	0.915/0.913	0.647
CAC [2]	√	×	0.985	0.938	0.803	0.863/0.872	0.772
OVRNs [16]	√	×	0.989	0.941	0.903	0.907/0.902	0.730
RPL [7]	√	×	0.996	0.968	0.901	0.976/0.968	0.809
A-RPL-CS [8]	√	×	0.997	0.967	0.910	0.971/0.951	0.782
Hybrid [12]	√	×	0.995	0.947	0.950	0.962/0.955	0.793
PROSER [3]	√	×	–	0.943	0.891	0.960/0.953	0.693
EGT [9]	√	×	–	0.958	0.821	0.937/0.930	0.709
Capsule [15]	√	×	0.992	0.956	0.835	0.888/0.889	0.715
S2OSC † [10]	×	√	0.995	0.936	0.855	0.910/0.809	0.714
A-RPL [8]	√	×	0.996	0.963	0.901	0.965/0.943	0.762
IT-OSR-ARPL	×	√	**0.999**	0.982	0.952	0.990/0.991	0.849
TransP	√	×	0.984	0.948	0.913	0.950/0.962	0.910
IT-OSR-TransP	×	√	**0.999**	**0.983**	**0.965**	**0.991/0.993**	**0.943**

- **TinyImageNet** [39]: TinyImageNet is a more complex dataset that contains 200 categories of ImageNet [40], 20 of which are randomly chosen as the known classes, while the rest 180 categories act as the unknown classes.

For a fair comparison, we use the same data splits as done in [6,9,12,13,19]. The details are presented in the supplementary material.

Under the cross-dataset setting, the whole 10 categories in the CIFAR10 dataset act as the known classes, while two datasets, TinyImageNet and LSUN [41], are either cropped or resized for acting as the unknown classes respectively, as done in [1,3–6,11,13,15,19].

4.1.2. Evaluation metrics

Under the standard-dataset setting, the AUROC and ACC are utilized as the evaluation metrics for evaluating the performance on detecting unknown classes and classifying known classes respectively as done in [1–9,11,13,15,19]:

- **AUROC:** The Receiver Operating Characteristic (ROC) Curve is depicted by the False Positive Rate (FPR) as abscissa and the True Positive Rate (TPR) as vertical coordinate. TPR is the ratio of known-class samples that are correctly predicted as known classes to all known-class samples, while FPR is the ratio of unknown-class samples that are erroneously predicted as known classes to all unknown-class samples. And the area under ROC curve (AUROC) is a typical evaluation metric for anomaly detection, which is not affected by the threshold chosen for separating between the two classes.
- **ACC:** The top-1 accuracy (ACC) is a typical evaluation metric in closed-set classification.

Under the cross-dataset setting, the macro-F1 score is utilized as the evaluation metric for evaluating the performance on classifying both known classes and unknown classes simultaneously as done in [1,3–6,11,13,15,19]:

- **macro-F1 score:** The macro-F1 score measures the $(c+1)$-way classification performance, which is not influenced by data imbalance.

4.2. Implementation details

In all of our experiments, we use the feature extractor in Swin-B [34] pretrained by ImageNet-22K [40] at $t = 1$, and use the feature extractor of M_t at $t > 1$ for the feature space. The α in Eq. (1) is set to 2.5, the number of nearest neighbors in Step S2 in the reliability sampling module is set to $K = 10$, the balancing weight λ in Eq. (6) is set to 0.1, and the maximum iteration number T is set to $T = 2$. In training, we use the SGD optimizer with the learning rate 0.002 for updating the feature extractor F of the baseline network, and 0.02 for updating the classifier part of the baseline network and training the generative network. The confidence score of the proposed inductive method TransP is defined as the maximum logit value outputted from the classifier. We use the conditional dual-adversarial generative network to generate a special number of features for known classes and unknown classes at each iteration during the training process as follows: (i) the number of the synthetic known-class features is equal to that of the real known-class features; (ii) the number of the sum of the real unknown-class features and the synthetic unknown-class features is equal to the sum of the real known-class features and the synthetic known-class features.

4.3. Evaluation under the standard-dataset setting

Aiming at both detecting unknown classes and classifying known classes, an effective method for open-set recognition should perform well on both the anomaly detection task and the closed-set classification task. Thus we compare the two derived transductive OSR methods with 19 open-set recognition methods on both anomaly detection and closed-set classification under the standard-dataset setting, and the results are reported in Tables 1 and 2 respectively. The results marked with † are reproduced by their published codes or by ourselves because these results are unreported in their papers.

As seen from Tables 1 and 2, compared with the two baseline inductive methods (A-RPL [8] and TransP), the performances of the two derived transductive OSR methods under the proposed IT-OSR framework are significantly improved on the relatively difficult datasets (*e.g.* TinyImageNet), indicating that the proposed transductive OSR framework is able to boost the performances of inductive OSR methods. Besides, the two derived transductive methods perform better than other 19 existing OSR methods for both anomaly detection and closed-set classification in most cases, demonstrating the effectiveness of the proposed framework for handling the open-set recognition task.

Table 2
Evaluation on closed-set classification (ACC) under the standard-dataset setting. The reported results are averaged over the same five trials as [6,9,12,13].

Method	Inductive	Transductive	MNIST	SVHN	CIFAR10	CIFAR+10/+50	TinyImageNet
SoftMax/ OpenMax [1]	√	×	0.995	0.947	0.801	–	–
OSRCI [13]	√	×	0.996	0.951	0.821	–	–
CROSR [4]	√	×	0.992	0.945	0.930	–	–
C2AE † [5]	√	×	0.992	0.936	0.910	0.919	0.430
CGDL † [11]	√	×	0.996	0.942	0.912	0.914	0.445
GCPL [17]	√	×	0.995	0.963	0.937	0.945	0.643
GMVAE [18]	√	×	0.996	0.962	0.946	0.952	0.729
MoEP-AE [19]	√	×	0.996	0.979	0.958	0.978	0.732
GDFR [6]	√	×	–	0.973	0.951	0.974	0.559
CAC [2]	√	×	0.998	0.970	0.934	0.952	0.759
OVRNs [16]	√	×	0.998	0.975	0.932	–	–
RPL † [7]	√	×	0.996	0.967	0.939	0.943	0.642
A-RPL-CS † [8]	√	×	0.997	0.971	0.953	0.956	0.678
Hybrid † [12]	√	×	0.995	0.962	0.926	0.937	0.612
PROSER [3]	√	×	–	0.964	0.926	–	0.521
EGT [9]	√	×	–	0.977	0.943	0.959	0.656
Capsule † [15]	√	×	0.994	**0.984**	0.952	0.969	0.774
S2OSC † [10]	×	√	0.996	0.941	0.925	0.918	0.689
A-RPL [8]	√	×	0.996	0.971	0.952	0.955	0.679
IT-OSR-ARPL	×	√	0.996	0.972	0.953	0.955	0.785
TransP	√	×	**0.997**	0.980	**0.988**	0.987	0.937
IT-OSR-TransP	×	√	**0.997**	0.980	**0.988**	**0.988**	**0.945**

Table 3
Evaluation on open-set classification (macro-F1 score) under the cross-dataset setting.

Dataset	Inductive	Transductive	ImageNet-crop	ImageNet-resize	LSUN-crop	LSUN-resize
SoftMax	√	×	0.639	0.653	0.642	0.647
OpenMax [1]	√	×	0.660	0.684	0.657	0.668
OSRCI [13]	√	×	0.636	0.635	0.650	0.648
CROSR [4]	√	×	0.721	0.735	0.720	0.749
C2AE [5]	√	×	0.837	0.826	0.783	0.801
CGDL [11]	√	×	0.840	0.832	0.806	0.812
GCPL [17]	√	×	0.807	0.793	0.829	0.795
GMVAE [18]	√	×	0.833	0.815	0.837	0.829
MoEP-AE [19]	√	×	0.858	0.841	0.889	0.875
GDFR [6]	√	×	0.757	0.792	0.751	0.805
CAC † [2]	√	×	0.764	0.752	0.756	0.777
OVRNs [16]	√	×	0.835	0.825	0.846	0.839
RPL † [7]	√	×	0.811	0.810	0.846	0.820
A-RPL-CS † [8]	√	×	0.862	0.841	0.859	0.873
Hybrid † [12]	√	×	0.802	0.786	0.790	0.757
PROSER [3]	√	×	0.849	0.824	0.867	0.856
EGT † [9]	√	×	0.829	0.794	0.826	0.803
Capsule [15]	√	×	0.857	0.834	0.868	0.882
S2OSC † [10]	×	√	0.828	0.810	0.832	0.806
A-RPL † [8]	√	×	0.858	0.830	0.845	0.867
IT-OSR-ARPL	×	√	0.939	0.910	0.921	0.908
TransP	√	×	0.881	0.870	0.908	0.891
IT-OSR-TransP	×	√	**0.971**	**0.959**	**0.973**	**0.971**

4.4. *Evaluation under the cross-dataset setting*

Comparing with the standard-dataset setting, there is a larger domain shift under the cross-dataset setting because the known classes and the unknown classes are from different datasets. We evaluate the two derived transductive OSR methods and the 19 existing OSR methods respectively under the cross-dataset setting, and the corresponding results are reported in Table 3, where those marked with † are reproduced.

As seen from Table 3, both of the derived IT-OSR methods perform significantly better than their baseline inductive methods as well as the other comparative methods. These results demonstrate that the proposed IT-OSR framework is insensitive to the relatively larger domain shift and it has a better generalization ability. This is probably because of the designed dual-space consistent sampling approach and the proposed conditional dual-adversarial generative network.

Table 4
OSR results on the CUB dataset from the Semantic Shift Benchmark.

Method	Inductive	Transductive	ACC	AUROC (Easy/Hard)	OSCR (Easy/Hard)
ARPL	√	×	0.952	0.948/0.844	0.912/0.819
TransP	√	×	0.949	0.945/0.845	0.915/0.822
S2OSC	×	√	0.948	0.931/0.829	0.897/0.803
IT-OSR-ARPL	×	√	0.961	0.960/0.858	0.924/0.834
IT-OSR-TransP	×	√	0.956	0.960/0.863	0.931/0.847

4.5. *Evaluation on fine-grained datasets*

The above results have demonstrated the effectiveness of IT-OSR-TransP on coarse-grained dataset. Here, we evaluate the following methods on both the CUB and FGVC-Aircraft datasets from the Semantic Shift Benchmark [42]:

Table 5
OSR results on the FGVC-Aircraft dataset from the Semantic Shift Benchmark.

Method	Inductive	Transductive	ACC	AUROC (Easy/Hard)	OSCR (Easy/Hard)
ARPL	√	×	0.915	0.880/0.784	0.836/0.749
TransP	√	×	0.902	0.855/0.778	0.813/0.736
S2OSC	×	√	0.908	0.852/0.763	0.806/0.715
IT-OSR-ARPL	×	√	0.920	0.904/0.816	0.857/0.773
IT-OSR-TransP	×	√	0.917	0.889/0.811	0.845/0.769

Table 6
Comparison of AUROC and ACC results by IT-OSR-TransP with $T = 1, 2, \ldots, 10$ on SVHN and CIFAR10 under the standard-dataset setting.

Dataset	SVHN		CIFAR10	
Metric	AUROC	ACC	AUROC	ACC
baseline	0.948	0.980	0.913	0.988
T = 1	0.979	0.980	0.963	0.987
T = 2	0.983	0.979	0.965	0.987
T = 3	0.984	0.977	0.968	0.985
T = 4	0.983	0.976	0.966	0.986
T = 5	0.983	0.977	0.961	0.986
T = 6	0.982	0.977	0.961	0.986
T = 7	0.983	0.976	0.969	0.985
T = 8	0.982	0.975	0.964	0.985
T = 9	0.982	0.976	0.965	0.984
T = 10	0.981	0.975	0.964	0.984

(i) Two inductive methods: the two proposed inductive baseline methods (including the existing ARPL [8] and the proposed TransP);

(ii) Three transductive methods: S2OSC [10], and the two transductive methods derived from the proposed IT-OSR framework by utilizing the above inductive methods as baseline methods (*i.e.*, IT-OSR-ARPL and IT-OSR-TransP).

The corresponding results are reported in Tables 4 and 5 respectively. As seen from the two tables, the two derived transductive methods outperform significantly their corresponding baseline inductive methods and the transductive method S2OSC [10]. These results are consistent with those on the basic datasets (MNIST, SVHN, CIFAR10, CIFAR+10/+50, TinyImageNet, ImageNet-crop, ImageNet-resize, LSUN-crop, and LSUN-resize) reported in Tables 1–3, demonstrating the effectiveness of the proposed transductive framework IT-OSR.

4.6. Analysis of the maximum iteration number T

In fact, the maximum number of iterations T depends on the number of the selected test samples at each iteration. Under the used selection strategy of test samples with replacement as described in Section 3, we find that the number of the selected test samples is generally close to or even more than $1/2$ of the whole test set at each iteration. Hence, after a few iterations, the model discriminability tends to converge, and most of the reliable test samples have been selected one time or more times, but some difficult test samples have never be selected at all times. Furthermore, we evaluate IT-OSR-TransP with $T = 1, 2, \ldots, 10$ on two datasets (SVHN and CIFAR10) under the standard-dataset setting. The corresponding results are reported in Table 6. As seen from this table, the results with $T = 2$ achieve a trade-off between accuracy and efficiency. Hence, we set $T = 2$ in all our experiments.

4.7. Ablation study

Table 7
Comparison of OSR results for ablation study on the orthogonal condition under both standard-dataset setting (TinyImageNet) and cross-dataset setting (ImageNet-crop).

Method	Standard-dataset setting		Cross-dataset setting
	AUROC	ACC	macro-F1
IT-OSR-TransP-HPE	0.932	0.941	0.958
IT-OSR-TransP	0.943	0.945	0.971

Table 8
Comparison of macro-F1 score for ablation study on DSCS and DGAN.

Method	macro-F1
TransP	0.881
IT-OSR-TransP w/o DSCS nor DGAN	0.949
IT-OSR-TransP w/o DGAN	0.964
IT-OSR-TransP w/o DSCS	0.962
IT-OSR-TransP with S2OSC-sampling	0.965
IT-OSR-TransP w/o feature space	0.967
IT-OSR-TransP w/o output space	0.966
IT-OSR-TransP	0.971

Ablation study on the orthogonal condition. In the proposed feature generation module, the known-class and unknown-class synthetic features are generated based on the $(C + 1)$-dimensional one-hot vectors, where the unknown-class condition is orthogonal to known-class conditions. Here, to better analyze the effect of such an orthogonal condition on the feature generation, we replace the proposed conditional generation setup from Fig. 2 with a setup with c classes, where synthetic negative examples are trained to produce high predictive entropy (similar to [28–30]) (called IT-OSR-TransP-HPE) in the transductive setup. The corresponding OSR results on TinyImageNet and on ImageNet-crop are reported in Table 7. As seen from this table, IT-OSR-TransP performs better than IT-OSR-TransP-HPE, demonstrating the effectiveness of the orthogonal condition.

Ablation study on the proposed sampling approach and generative network. Then, to better analyze the effect of the explored dual-space consistent sampling approach (denoted DSCS) and the proposed conditional dual-adversarial generative network (denoted DGAN), we conduct an ablation study on DSCS and DGAN. The experiment is implemented in the IT-OSR-TransP method on the ImageNet-crop dataset under the cross-dataset setting, where we compare the methods not only with or without (denoted w/o) DSCS and DGAN but also with the sampling approach in S2OSC [10] (denoted S2OSC-sampling) instead of DSCS, and the results are reported in Table 8. Note that the methods w/o DSCS sample from the test set only by a scoring approach (*i.e.*, the Step S1 in the reliability sampling module), the methods w/o DGAN train a typical conditional GAN (*i.e.*, $G + F + D_1 + C_1$) instead in the feature generation module. As seen from Table 8, the comparison between IT-OSR-TransP w/o DGAN and IT-OSR-TransP demonstrates the effect of the proposed DGAN, and the comparisons of IT-OSR-TransP to both IT-OSR-TransP w/o DSCS and IT-OSR-TransP with S2OSC-sampling demonstrate that the explored DSCS is effective. It is noted that the sample imbalance problem is a common problem in the OSR task and many other visual tasks. Since the proposed model could generate synthetic features, it could alleviate the sample imbalance problem for improving the accuracy of open-set recognition. Moreover, due to the fact that the proposed module could generate various features by utilizing different noises, the variety of synthetic negative features may be also helpful for improving the accuracy of open-set recognition to some extent.

Ablation study on the two spaces in the proposed sampling approach. Here, we evaluate the separate influence of the two spaces in the proposed pseudo-labeling selection criterion (*i.e.*, the output space for obtaining confidence scores and the feature space for obtaining feature distances) in the reliability sampling module. Specifically, we modify IT-OSR-TransP into the following two methods for comparison:

(i) IT-OSR-TransP w/o feature space: Here, we remove the sampling process in the feature space from the proposed sampling approach, such that the test samples are sampled only based on their confidence scores in the output space.

(ii) IT-OSR-TransP w/o output space: Here, we remove the process of assigning pseudo labels in the output space from the proposed sampling approach, such that the test samples are sampled only based on the feature distances in the feature space (the minimum distance between a test feature to each known-class feature center).

The OSR results under the cross-dataset setting (ImageNet-crop) are reported in Table 8. As seen from this table, both IT-OSR-TransP w/o feature space and IT-OSR-TransP w/o output space perform better than TransP, indicating that both single-space sampling approaches could sample reliable test samples to different extents. In fact, IT-OSR-TransP w/o feature space adopts a traditional sampling approach, which selects samples with large logit values by the corresponding updated model at each iteration. However, due to the limitation of the model discriminability, the pseudo labels of some test samples selected by the traditional sampling approach are inevitably wrong. Addressing this problem, the proposed dual-space consistent sampling approach further selects test samples whose pseudo labels are consistent with those of most their neighbors in the feature space. Hence, the reliability sampling module is able to improve the sample selection so that the accuracy of open-set recognition is improved. Moreover, we have compared the sampling accuracy of the traditional sampling strategy with that of the proposed sampling approach, finding that the sampling accuracy of the proposed sampling approach (about 90%) is higher than that of the traditional sampling strategy (about 70%). Besides, IT-OSR-TransP further improves the performance of both IT-OSR-TransP w/o feature space and IT-OSR-TransP w/o output space, demonstrating the effectiveness of the proposed dual-space consistent sampling approach.

Reduced Sample Imbalance vs. Variety of Negative Data. The above results have demonstrated the effectiveness of the proposed feature generation approach. Here, we conduct experiments on TinyImageNet to analyze the effects of both the sample imbalance and the variety of the negative samples on the experimental advantage of the proposed approach as follows:

We use the 8000 training samples from TinyImageNet as positive samples, and design the following three settings on different ratios R_{pn} of positive samples to negative features for evaluating the effect of the sample imbalance:

(i) $R_{pn} = 10 : 1$, where the number of the generated negative features is 800.

(ii) $R_{pn} = 4 : 1$, where the number of negative features is 2000.

(iii) $R_{pn} = 1 : 1$, where the number of negative features is 8000.

Then, under each configuration of the above ratios of positive samples to negative features, we use different numbers N_{noise} of random noises (here, $N_{noise} = 800, 300, 100$) to synthesize negative samples for evaluating the effect of the variety of the negative sample.

Under different combinations of R_{pn} and N_{noise}, the proposed method IT-OSR-TransP is trained respectively. Then it is evaluated under both standard-dataset setting (TinyImageNet) and cross-dataset setting (ImageNet-crop), and the results are reported in Table 9. The following two points could be observed from this table:

(i) Under each configuration of the ratio R_{pn} of positive samples to negative features, the performance of the proposed method becomes better when the noise number N_{noise} increases (*i.e.*, the negative samples become more various). This demonstrates that the variety of the negative samples is one cause of the advantage of the proposed method.

(ii) Under each configuration of the noise number N_{noise}, the performance of the proposed method becomes better when the ratio R_{pn} of positive samples to negative features gets closer to 1 (*i.e.*, there is a balance between the numbers of positive and negative samples). This demonstrates that the reduced sample imbalance is also one cause of the advantage of the proposed method.

In sum, both the reduced sample imbalance and the variety of the negative samples are the causes of the advantage of the proposed method.

Table 9
OSR results of IT-OSR-TransP with different sample ratios R_{pn} and different noise numbers N_{noise} under both standard-dataset setting (TinyImageNet) and cross-dataset setting (ImageNet-crop).

R_{pn}	N_{noise}	Standard-dataset setting		Cross-dataset setting
		AUROC	ACC	macro-F1
10:1	800	0.929	0.939	0.935
	300	0.923	0.937	0.922
	100	0.915	0.937	0.896
4:1	800	0.930	0.939	0.940
	300	0.926	0.938	0.925
	100	0.919	0.937	0.901
1:1	800	0.933	0.940	0.949
	300	0.928	0.938	0.928
	100	0.921	0.938	0.906

Table 10
Comparison of macro-F1 score for ablation study on D_2 and C_2 of the proposed DGAN.

Architecture	macro-F1
$G + F + D_1 + C_1$	0.964
$G + F + D_1 + C_1 + C_2$	0.966
$G + F + D_1 + D_2 + C_1$	0.969
IT-OSR-TransP	0.971

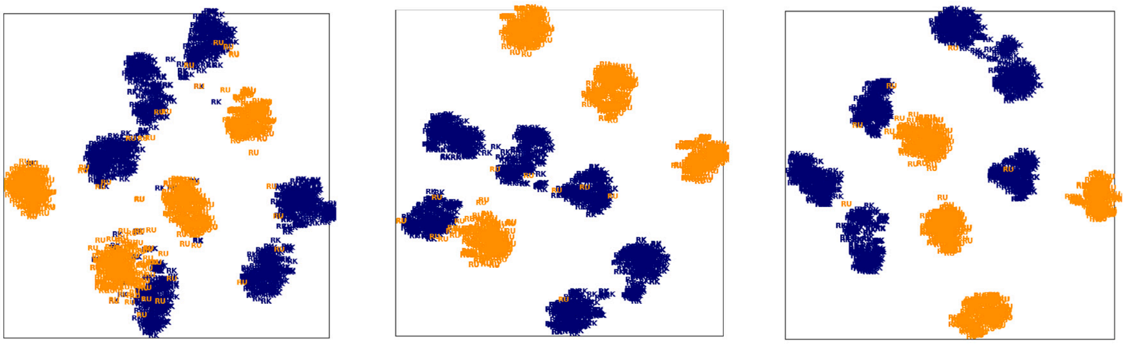
Ablation study on the architecture of the proposed generative network. Although the classifier C_2 and the discriminator D_2 seem to handle a same task: classifying/discriminating the features as either known classes or unknown classes, there exists two differences. One difference lies in the outputs. C_2 outputs a 2-dimensional vector, while D_2 outputs a single value. The other difference lies in the training losses. C_2 is trained with the cross-entropy classification loss, while D_2 is trained with the adversarial loss. Further, to better analyze the effect of C_2 and D_2 in the proposed DGAN, we conduct an ablation study on D_2 and C_2 (G, F, D_1 and C_1 are indispensable), which is also implemented in the IT-OSR-TransP method on the ImageNet-crop dataset under the cross-dataset setting. The results are reported in Table 10. As seen from Table 10, the comparison between $G + F + D_1 + C_1 + C_2$ and IT-OSR-TransP demonstrates that the discriminator D_2 in the proposed DGAN is important for improving the discrimination of the known/unknown-class features, and the comparison between $G + F + D_1 + D_2 + C_1$ and IT-OSR-TransP demonstrates that the classifier C_2 in the proposed DGAN is also effective for better performance.

4.8. Visualization

In this section, we adopt t-SNE to visualize the extracted features from both the real known-class and unknown-class images by IT-OSR-TransP at three iterations $t = 0, 1, 2$ (IT-OSR-TransP with $t = 0$ is indeed the initial baseline inductive method TransP) on the CIFAR10 dataset in Fig. 3. In this figure, the real known-class features and real unknown-class features are denoted as blue 'RK' and yellow 'RU', respectively.

As seen from Fig. 3(a), some extracted unknown-class features are overlapped with extracted known-class features, indicating that the feature collapse problem does exist in the baseline feature extractor. As seen from Fig. 3(b) and (c), the overlapped regions have been reduced progressively with the increase of the iteration number t, demonstrating that the feature collapse problem could be alleviated by the proposed method.

Besides, for verifying whether the proposed method suffers from mode collapse or not, we use t-SNE to visualize the generated features as well as the real features on CIFAR10 by the proposed method in Fig. 4. In this figure, the generated known-class features and generated unknown-class features are denoted as green 'GK' and red 'GU', respectively. As seen from this figure, the generated features generally cover



(a) (b) (c)

Fig. 3. T-SNE visualization of the known-class real features (denoted as blue 'RK') and unknown-class real features (denoted as yellow 'RU') extracted by the updated feature extractor F of (a) IT-OSR-TransP with $t = 0$ (*i.e.*, TransP), (b) IT-OSR-TransP with $t = 1$, and (c) IT-OSR-TransP with $t = 2$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

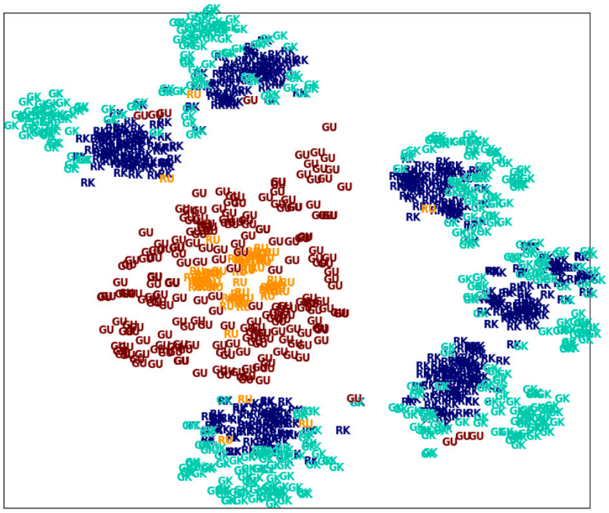


Fig. 4. T-SNE visualization of the known-class real features (denoted as blue 'RK') and unknown-class real features (denoted as yellow 'RU') as well as the known-class generated features (denoted as green 'GK') and unknown-class generated features (denoted as red 'GU'). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

the real distributions, demonstrating that the proposed method does not suffer from the mode collapse problem heavily.

5. Conclusion

In this paper, we propose the Iterative Transductive OSR framework, IT-OSR, which iteratively performs three explored modules: the reliability sampling module, the feature generation module, and the baseline update module. In the reliability sampling module, we explore the dual-space consistent sampling approach for selecting a relatively reliable subset from the pseudo-labeled test samples. In the feature generation module, we explore the conditional dual-adversarial generative network for feature generation, which could balance the number of the pseudo-labeled unknown-class test samples and that of the known-class samples. Any inductive OSR method can be seamlessly embedded into IT-OSR for alleviating the domain shift problem. We further derive two novel transductive OSR methods under the explored IT-OSR framework, and extensive experimental results demonstrate the effectiveness of IT-OSR.

Although the proposed feature generation module has shown its effectiveness for handling the OSR task, the orthogonal condition currently used in this model still leaves much room for improvement. In fact, the orthogonal-coding strategy is tentative, and the features of unknown-class samples are not necessarily orthogonal to those of known-class samples in some latent feature spaces. Hence, a more effective coding condition would be expected to boost the performance of the feature generation module further, which would be one of our future works.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The code is available at: https://github.com/sjy-1995/IT-OSR-code.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant No. 61991423, U1805264, 61972374), the Strategic Priority Research Program of the Chinese Academy of Sciences, China (Grant No. XDA27040811), the Beijing Municipal Science and Technology Project, China (Grant No. Z211100011021004).

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.patcog.2023.110046.

References

[1] A. Bendale, T.E. Boult, Towards open set deep networks, in: IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 1563–1572.

[2] D. Miller, N. Sünderhauf, M. Milford, F. Dayoub, Class anchor clustering: A loss for distance-based open set recognition, in: IEEE Workshop on Applications of Computer Vision, 2021, pp. 3570–3578.

[3] D. Zhou, H. Ye, D. Zhan, Learning placeholders for open-set recognition, in: IEEE Conference on Computer Vision and Pattern Recognition, 2021, pp. 4401–4410.

[4] R. Yoshihashi, W. Shao, R. Kawakami, S. You, M. Iida, T. Naemura, Classification-reconstruction learning for open-set recognition, in: IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 4016–4025.

[5] P. Oza, V.M. Patel, C2ae: Class conditioned auto-encoder for open-set recognition, in: IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 2307–2316.

[6] P. Perera, V.I. Morariu, R. Jain, V. Manjunatha, C. Wigington, V. Ordonez, V.M. Patel, Generative-discriminative feature representations for open-set recognition, in: IEEE Conference on Computer Vision and Pattern Recognition, 2020, pp. 11814–11823.

[7] G. Chen, L. Qiao, Y. Shi, P. Peng, J. Li, T. Huang, S. Pu, Y. Tian, Learning open set network with discriminative reciprocal points, in: European Conference on Computer Vision, 2020, pp. 507–522.

[8] G. Chen, P. Peng, X. Wang, Y. Tian, Adversarial reciprocal points learning for open set recognition, in: IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 44, No. 11, 2017, pp. 8065–8081.

[9] P. Perera, V.M. Patel, Geometric transformation-based network ensemble for open-set recognition, in: IEEE International Conference on Multimedia & Expo, 2021.

[10] Y. Yang, H. Wei, Z. Sun, G. Li, Y. Zhou, H. Xiong, J. Yang, S2OSC: A holistic semi-supervised approach for open set recognition, ACM Trans. Knowl. Discov. Data 16 (34) (2022) 1–27.

[11] X. Sun, Z. Yang, C. Zhang, K.-V. Ling, G. Peng, Conditional Gaussian distribution learning for open set recognition, in: IEEE Conference on Computer Vision and Pattern Recognition, 2020, pp. 13480–13489.

[12] H. Zhang, A. Li, J. Guo, Y. Guo, Hybrid models for open set recognition, in: European Conference on Computer Vision, 2020, pp. 102–117.

[13] L. Neal, M. Olson, X. Fern, W.-K. Wong, F. Li, Open set learning with counterfactual images, in: European Conference on Computer Vision, 2018, pp. 613–628.

[14] S. Kong, D. Ramanan, OpenGAN: Open-set recognition via open data generation, in: IEEE International Conference on Computer Vision, 2021, pp. 813–822.

[15] Y. Guo, G. Camporese, W. Yang, A. Sperduti, L. Ballan, Conditional variational capsule network for open set recognition, in: IEEE International Conference on Computer Vision, 2021, pp. 103–111.

[16] J. Jang, C.O. Kim, Collective decision of one-vs-rest networks for open-set recognition, IEEE Trans. Neural Netw. Learn. Syst. (early access) (2022) http://dx.doi.org/10.1109/TNNLS.2022.3189996.

[17] H. Yang, X. Zhang, F. Yin, C. Liu, Robust classification with convolutional prototype learning, in: IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 3474–3482.

[18] A. Cao, Y. Luo, D. Klabjan, Open-set recognition with Gaussian mixture variational autoencoders, in: AAAI Conference on Artificial Intelligence, Vol. 35, No. 8, 2021, pp. 6877–6884.

[19] J. Sun, H. Wang, Q. Dong, MoEP-AE: Autoencoding mixtures of exponential power distributions for open-set recognition, IEEE Trans. Circuits Syst. Video Technol. 33 (1) (2022) 312–325, http://dx.doi.org/10.1109/TCSVT.2022.3200112.

[20] E. Lopez-Lopez, X.M. Pardo, C.V. Regueiro, Incremental learning from low-labelled stream data in open-set video face recognitions, Pattern Recognit. 131 (2022) http://dx.doi.org/10.1016/j.patcog.2022.108885.

[21] T.G. Dietterich, A. Guyer, The familiarity hypothesis: Explaining the behavior of deep open set methods, Pattern Recognit. 132 (2022) http://dx.doi.org/10.1016/j.patcog.2022.108931.

[22] H. Shao, D. Zhong, Towards open-set touchless palmprint recognition via weight-based meta metric learning, Pattern Recognit. 121 (2022) http://dx.doi.org/10.1016/j.patcog.2021.108247.

[23] H. Cevikalp, B. Uzun, O. Köpüklü, G. Ozturk, Deep compact polyhedral conic classifier for open and closed set recognition, Pattern Recognit. 119 (2021) http://dx.doi.org/10.1016/j.patcog.2021.108080.

[24] B. Liu, Q. Dong, Z. Hu, Hardness sampling for self-training based transductive zero-shot learning, in: IEEE Conference on Computer Vision and Pattern Recognition, 2021, pp. 16499–16508.

[25] B. Liu, Q. Dong, Z. Hu, An iterative co-training transductive framework for zero-shot learning, IEEE Trans. Image Process. 30 (2021) 6943–6956.

[26] H. Chen, Y. Yu, Y. Jia, B. Gu, Incremental learning for transductive support vector machine, Pattern Recognit. 133 (2023) http://dx.doi.org/10.1016/j.patcog.2022.108982.

[27] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, Commun. ACM 63 (11) (2020) 139–144.

[28] K. Lee, H. Lee, K. Lee, J. Shin, Training confidence-calibrated classifiers for detecting out-of-distribution samples, in: International Conference on Learning Representations, 2018.

[29] M. Grcić, P. Bevandić, S. Šegvić, Dense open-set recognition with synthetic outliers generated by real NVP, in: International Conference on Computer Vision Theory and Applications, 2021.

[30] Z. Zhao, L. Cao, K.-Y. Lin, Revealing the distributional vulnerability of discriminators by implicit generators, 2021, arXiv preprint. arXiv:2108.09976.

[31] X. Du, Z. Wang, M. Cai, Y. Li, VOS: Learning what you don't know by virtual outlier synthesis, in: International Conference on Learning Representations, 2022.

[32] A. Odena, C. Olah, J. Shlens, Conditional image synthesis with auxiliary classifier GANs, in: Proceedings of the 34th International Conference on Machine Learning, Vol. 70, 2017, pp. 2642–2651.

[33] A. Martin, S. Chintala, L. Bottou, Wasserstein generative adversarial networks, in: International Conference on Machine Learning, 2017, pp. 214–223.

[34] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, B. Guo, Swin transformer: Hierarchical vision transformer using shifted windows, in: IEEE International Conference on Computer Vision, 2021, pp. 10012–10022.

[35] Y. LeCun, C. Cortes, C. Burges, Mnist handwritten digit database, 2010, Available: http://yann.lecun.com/exdb/mnist.

[36] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, A.Y. Ng, Reading digits in natural images with unsupervised feature learning, in: Conference and Workshop on Neural Information Processing Systems, 2011.

[37] A. Krizhevsky, G. Hinton, Convolutional deep belief networks on cifar-10, 2010, *Unpublished manuscript*.

[38] A. Krizhevsky, G. Hinton, Learning multiple layers of features from tiny images, Tech. rep, 2009.

[39] Y. Le, X. Yang, Tiny imagenet visual recognition challenge, 2015, *CS 231N*.

[40] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, C.B. Alexander, F. Li, Imagenet large scale visual recognition challenge, Int. J. Comput. Vision 115 (3) (2015) 211–252.

[41] F. Yu, A. Seff, Y. Zhang, S. Song, T. Funkhouser, J. Xiao, LSUN: Construction of a large-scale image dataset using deep learning with humans in the loop, 2015, arXiv preprint. arXiv:1506.03365.

[42] S. Vaze, K. Han, A. Vedaldi, A. Zisserman, The semantic shift benchmark, in: ICML 2022 Workshop Shift Happens, 2022.

Jiayin Sun is pursuing the Ph.D. in pattern recognition and intelligence systems with the NLPR, CASIA. Her current research interests include pattern recognition and its applications, particularly in open-set recognition.

Qiulei Dong received the Ph.D. degree from CASIA in 2008. He is currently a Professor with the NLPR, CASIA, an Adjunct Professor in the UCAS, and at the CEBSIT, CAS. He serves as a Young Associate Editor of the Journal of Computer Science and Technology. His current research interests include 3-D computer vision, pattern classification, and biological-vision-based modeling.