Progressively Select and Reject Pseudo-labelled Samples for Open-Set Domain Adaptation

Qian Wang, Fanlin Meng, *Senior Member, IEEE,* Toby P. Breckon, *Senior Member, IEEE,*

Abstract—Domain adaptation solves image classification problems in the target domain by taking advantage of the labelled source data and unlabelled target data. Usually, the source and target domains share the same set of classes. As a special case, Open-Set Domain Adaptation (OSDA) assumes there exist additional classes in the target domain but are not present in the source domain. To solve such a domain adaptation problem, our proposed method learns discriminative common subspaces for the source and target domains using a novel Open-Set Locality Preserving Projection (OSLPP) algorithm. The source and target domain data are aligned in the learned common spaces class-wise. To handle the open-set classification problem, our method progressively selects target samples to be pseudo-labelled as known classes, rejects the outliers if they are detected as unknown classes, and leaves the remaining target samples as uncertain. The common subspace learning algorithm OSLPP simultaneously aligns the labelled source data and pseudo-labelled target data from known classes and pushes the rejected target data away from the known classes. The common subspace learning and the pseudo-labelled sample selection/rejection facilitate each other in an iterative learning framework and achieve state-of-the-art performance on four benchmark datasets Office-31, Office-Home, VisDA17 and Syn2Real-O with the average HOS of 87.6%, 67.0%, 76.1% and 65.6% respectively.

Impact Statement—Traditional supervised machine learning methods require a large amount of labelled data for model training. In many real-world applications, however, collecting and labelling sufficient data is laborious and even impossible. One way to solve this problem is domain adaptation which solves problems (e.g., image classification) in the target domain by taking advantage of the labelled data in a different but related domain (i.e. source domain). This paper aims to address the open-set domain adaptation problem in image classification. Our proposed approach relies on deep features and a feature transformation method so that it can not only achieve state-of-the-art performance on benchmark datasets but also is more practically useful due to its low training cost. Although we demonstrate the effectiveness of our proposed approach in the image classification tasks, we have found it also applicable in many other domains such as drug discovery and computational biology. We believe this work will benefit the broad community of AI in a variety of applications.

Index Terms—Open-set domain adaptation, Pseudo-labelling, Locality preserving projection

I. INTRODUCTION

Q. Wang is with the Department of Computer Science, Durham University, United Kingdom, e-mail: qian.wang173@hotmail.com

F. Meng is with the Alliance Manchester Bussiness School, The University of Manchester, UK

TP. Breckon is with the Department of Computer Science and Department of Engineering, Durham University, United Kingdom, e-mail: toby.breckon@durham.ac.uk

ONE key to modern learning systems is the access to large-scale high-quality training data. Collecting and annotating a large amount of data for model training can be difficult and costly in some particular domains [1], [2], [3]. Given a target domain where the labelled training data are limited, one may promote the learning by exploiting annotated data from a source domain where annotated data are easier to access. For instance, object recognition from photos taken at night as a target domain task can be better addressed by exploiting more easily accessible photos taken in the daytime (i.e. source domain). However, the data distribution shift between the source and the target domains degrades the traditional transfer learning performance. To address the domain shift problem, domain adaptation techniques have been extensively studied in recent years. In particular, effective approaches have been proposed to solve the closed set domain adaptation problems in which the source and target domain share the same set of classes [4], [5], [6], [7]. In many real-world applications, however, we may only have interest in a subset of the classes in the target domain and ignore the rest of them. In other cases, the annotated data in the source domain may not necessarily cover all classes in the target domain. Arising from such realistic scenarios, the open-set domain adaptation (OSDA) problem aiming to recognize the target-domain samples as one of the known classes (i.e. shared classes between the source and target domains) or the unified unknown class, has attracted much attention [8], [9], [10], [11], [12], [13], [14], [15].

Typical closed-set domain adaptation methods suffer from the negative transfer issue [16] when directly applied to the OSDA problems. Specifically, most closed-set domain adaptation methods take advantage of the prior knowledge that the source and target domain share the same set of classes and conditional distributions can be well aligned. With the existence of unknown classes in the target domain for OSDA, data belonging to these unknown classes will be mistakenly aligned with some known classes in the source domain. Many existing OSDA methods attempt to solve this problem by treating the unknown classes as a unified one and learning a classifier for $C+1$ classes [9], [15], [12], [10]. Such a learning objective forces different unknown classes to behave similarly (e.g., clustered compactly in the hidden representation space), increasing the difficulty of learning a performant classifier. We believe that such a learning object is not necessary for an OSDA problem which aims to discriminate each known class from the rest. In contrast, we solve the OSDA problem by learning a common subspace where known classes are separated from each other as well as all the unknown classes

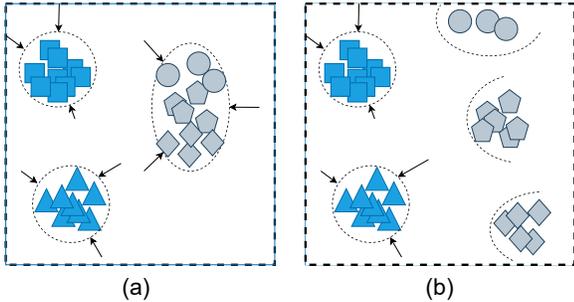whilst the unknown classes are not necessarily compactly clustered.

Fig. 1. An illustration of the motivations of our proposed approach (marker shapes represent different classes; known classes and unknown classes are represented by the blue and grey colours respectively; the arrows represent the explicitly optimized objectives during subspace learning). (a) Existing approaches tend to push all the unknown classes together when learning the $(C+1)$−class classifier; (b) we relax the constraint by only pushing the known classes together and the remaining unknown classes will be relatively far away from these known class clusters.

To learn such a favourable common subspace from two domains, our proposed framework has two essential algorithms: (1) the algorithm of progressively selecting and rejecting pseudo-labelled samples for unknown classes aware discriminative learning and (2) the domain adaptation algorithm OSLPP (Open-Set Locality Preserving Projection), our modified version of LPP [17] for domain aligned common subspace learning. These two algorithms facilitate each other during favourable subspace learning for optimal open-set domain adaptation.

The first algorithm is adapted from a typical UDA approach [16] by adding an unknown class sample rejection process in each iteration. Specifically, in [16], the target domain data are pseudo-labelled as the known classes and progressively selected as supervision for the domain adaptation process. In our proposed method, to enable unknown classes aware discriminative learning, we additionally reject some pseudo-labelled target samples as unknown classes if these samples are far away from all the known classes so that the unknown classes can be considered during common subspace learning using OSLPP. The remaining target samples are treated as uncertain and hence will not contribute to the domain alignment. It is important to select and reject the pseudo-labelled samples progressively so that the negative effect caused by incorrect pseudo-labelling can be mitigated. To make the most of the pseudo-labelled target samples, the numbers of selected and rejected target samples are monotonically increasing until no samples are remaining uncertain.

The second algorithm aims at aligning the source and target domains in a learned common subspace in which the source and target data from the known classes are aligned class-wise whilst the target domain data pseudo-labelled as unknown classes are pushed away from the known classes. Inherited from the original LPP algorithm, the OSLPP has the capability of local structure preserving which is important for model generalization from the source domain to the target domain. Different from the original LPP, OSLPP is a supervised

learning algorithm taking advantage of the labelled source data and the pseudo-labelled target data. OSLPP also differs from the supervised LPP used in [16] by taking into consideration the estimated unknown classes when learning the projection. In addition, the relatively loose constraints of the OSLPP algorithm allow the unknown classes distributed in a spacious region while separated from known classes. The OSLPP-based subspace learning and pseudo-labelling are conducted alternately and repeated for a fixed number of iterations. These two algorithms facilitate each other iteratively and the two domains are well aligned in the learned subspace where the recognition performance of target domain data is enhanced.

The contributions of this work can be summarized as follows:

- A novel framework is proposed for open set domain adaptation by learning a common subspace from both source and target domains using OSLPP, a novel algorithm aiming at aligning data from known classes and pushing away data from unknown classes.
- An algorithm of progressively selecting and rejecting pseudo-labelled target domain data is proposed to facilitate the domain adaptation algorithm.
- Experiments are conducted on four commonly used datasets Office-31, Office-Home, VisDA17 and Syn2Real-O. The experimental results demonstrate our proposed method can achieve or outperform state-of-the-art performance.
- We empirically demonstrate that hyper-parameter values can trade off the accuracy of classifying known classes and the accuracy of detecting unknown classes in real-world OSDA problems.

II. RELATED WORK

In this section, we review existing related work in domain adaptation (including closed-set domain adaptation, partial domain adaptation and universal domain adaptation), open-set recognition and open-set domain adaptation.

A. Domain Adaptation

Domain adaptation is a general technique being used to address various research problems. From the perspective of tasks, it can be applied to image classification [18], [10], object detection [19], image segmentation [20], person re-identification [21], [22], etc. From the perspective of supervision in the *target* domain, it can be categorized into unsupervised, semi-supervised or supervised domain adaptation although most existing works including ours focus on the unsupervised setting. From the perspective of label space overlapping between two domains, it has been formulated into closed-set domain adaptation where the label spaces are equivalent between two domains, partial domain adaption (PDA) where the source domain contains extra classes [23], [24], [25], OSDA which is our focus in this work and universal domain adaptation where a versatile approach is expected to solve the problem without knowing it is a closed-set, open-set or partial domain adaptation problem [26], [27], [28], [29], [30], [29], [31].

Focusing on the image classification tasks, the OSDA problem we attempt to address is unsupervised in the sense that there is no labelled target data. Our work is inspired by the unsupervised domain adaptation method proposed in [16]. Using a similar iterative learning framework, our method is dedicated to OSDA problems by detecting the unknown classes in the target domain and exploiting them to facilitate the common subspace learning for domain adaptation.

B. Open-Set Domain Adaptation

Existing OSDA methods borrow the successful ideas from unsupervised domain adaptation approaches and adapt them for OSDA by handling the unknown target samples in specific ways. These methods are distinct from each other in how the unknown target samples are detected and utilized. Saito et al. [8] learn a classifier to classify target samples into $C + 1$ classes (C known classes and 1 unknown class) and use such pseudo-labels to construct the loss function for classifier training. Pan et al. [32] try to explore the structure of the target data by Self-Ensembling with Category-agnostic Clusters (SE-CC) to improve the recognition of unknown classes in the target domain. Liu et al. [9] take one step further by progressive adaptation with selected pseudo-labelled samples in the target domain. As this progressive learning algorithm has been proven effective for domain adaptation [16], our work adopts a similar progressive learning algorithm but utilizes novel strategies for data selection/rejection and OSLPP for domain alignment. Following the same direction, Fang et al. [12] use the samples classified as the unknown classes in the so-called *open set difference* loss term to enhance the ability to recognize the unknown classes of the learned classifier. However, this method introduces too many hyper-parameters making it difficult to use in practice.

Most aforementioned methods have coupled unknown class detection and domain adaptation modules, however, exceptions exist that address the OSDA problem in two stages [10], [33]. In the first stage of [10], a classifier for separating known and unknown classes is learned using rotation-based self-supervised learning. In the second stage, the source samples together with the detected known target samples are combined to train a classifier to classify target samples into either one of the known classes or the unknown class. The limitation of this method is that the two stages are trained sequentially, hence the performance of the second stage relies on the accuracy of known-unknown class separability in the first stage and may lead to a sub-optimal solution. Kundu et al. [33] aim at solving OSDA in a special scenario where source data are separated from the target data. Instead of training a model using the combination of source and target domain data, an inheritable model is first trained with the source data and subsequently adapted for the target data. The unknown classes are recognized by measuring the *instance-level inheritability*. In addition, the samples confidently pseudo-labelled as unknown classes are used as target-domain supervision. Our work follows a similar idea of selecting the most confident pseudo-labels as supervision information from the target domain and such information is expected to better cluster the

known classes and discriminate unknown classes far away from the known classes. The difference lies in that we use a novel manifold learning method OSLPP as opposed to the neural networks employed in [33] and achieve superior or comparable performance.

TABLE I
NOMENCLATURE.

Symbol	Meaning
$\mathcal{D}^s, \mathcal{D}^t$	the source, target dataset
$\boldsymbol{x}_i^s, \boldsymbol{x}_i^t$	the feature vector of i-th source, target sample
n_s, n_t	the number of source, target samples
$\mathcal{Y}^s, \mathcal{Y}^t, \mathcal{Y}^{unk}$	the label space of source, target, unknown data set
W	the adjacency matrix for LPP
P	the learned projection matrix by LPP
$\boldsymbol{z}^s, \boldsymbol{z}^t$	the projections of source, target samples

III. PROBLEM FORMULATION

Suppose we have a labelled data set $\mathcal{D}^s = \{(\boldsymbol{x}_i^s, y_i^s)\}, i = 1, 2, ..., n_s$ from the source domain $\mathcal{S}$, $\boldsymbol{x}_i^s \in \mathbb{R}^{d_0}$ represents the feature vector of i-th labelled sample in the source domain, d_0 is the feature dimension and $y_i^s \in \mathcal{Y}^s$ denotes the corresponding label. OSDA aims at classifying an unlabelled data set $\mathcal{D}^t = \{\boldsymbol{x}_i^t\}, i = 1, 2, ..., n_t$ from the target domain $\mathcal{T}$, where $\boldsymbol{x}_i^t \in \mathbb{R}^{d_0}$ represents the feature vector in the target domain. The target label space $\mathcal{Y}^t$ is a union of the source label space $\mathcal{Y}^s$ and an unknown label space $\mathcal{Y}^{unk}$ (i.e. $\mathcal{Y}^{unk} = \mathcal{Y}^t \backslash \mathcal{Y}^s \neq \emptyset$). The classes shared by the source and target domains are *known classes* whilst the rest of the target classes are unknown classes ($\mathcal{Y}^{unk}$). Samples from the target domain are expected to be classified as one of the known classes or the unified unknown class. It is assumed that both the labelled source domain data $\mathcal{D}^s$ and the unlabelled target domain data $\mathcal{D}^t$ are available during model training. To clarify, we will use the terms "select" and "reject" to represent the actions of confidently classifying/pseudo-labelling a target sample into a known class or the unknown class, respectively. The rest target samples which are neither "selected" nor "rejected" are denoted as "uncertain".

IV. METHOD

We introduce our proposed approach to the OSDA problems in this section. The framework of the approach is first described and illustrated in Figure2. The original version of the LPP algorithm is briefly described to make the paper self-contained. Subsequently, we give the details of two key algorithms in the framework: OSLPP and pseudo-label selection and rejection. Finally, the algorithm is summarized in Algorithm 1.

A. Overview

Our approach to OSDA is inspired by the existing approaches to unsupervised domain adaptation [34], [5], [16] and uses a similar iterative learning framework. The goal is to learn a common subspace based on the labelled source samples (c.f. Fig.2a) and pseudo-labelled target samples (c.f. Fig.2b)
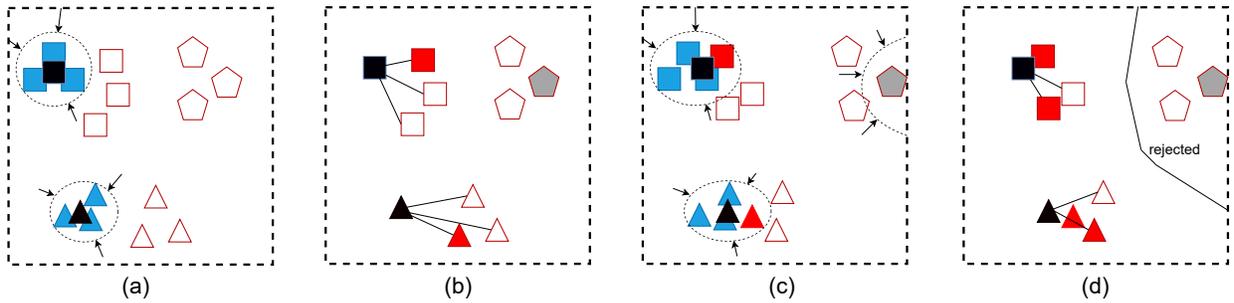
Fig. 2. An illustration of our proposed approach to OSDA. The blue and red colours are used to represent the source and target domains, respectively; different shaped markers represent different classes; the black markers are the class means; the hollow red markers are unlabelled target samples which are filled in with the red colour if selected as known classes and filled in with the grey colour if rejected as unknown classes. (a) A supervised LPP is applied to the labelled source data to pull the source samples from the same class closer to each other. (b) In the subspace learned by (a), class means are computed and used to selectively assign pseudo-labels to the target samples; on the other hand, target samples far away from all class means are rejected as unknown classes. (c) Our proposed OSLPP is applied to the labelled source samples, selected pseudo-labelled target samples and rejected target samples so that the samples from the same class regardless of their domains are pulled close to each other whilst those rejected as unknown classes are pushed away from the known classes. (d) In the subspace learned by (c), class means are computed and used to selectively assign pseudo-labels to the known classes; the target samples are rejected if their nearest neighbour is already rejected. Steps (c) and (d) are repeated for T times.

so that the source and target domains can be well aligned in the learned subspace and the target samples can be recognized by a simple nearest neighbour method (c.f. Fig.2c). It has been proved that selective pseudo-labelling performs better than the methods considering all the pseudo-labelled target samples without the selection [16]. The subspace learning procedure and the selective pseudo-labelling facilitate each other during the learning. Distinct from existing works, we propose a new subspace learning method OSLPP, an extension of LPP to the OSDA problems, and a novel selection-rejection algorithm to allow the OSLPP to learn a better subspace where known classes are aligned and unknown classes are pushed away (c.f. Fig.2d).

B. Locality Preserving Projection

To make the paper self-contained, we briefly describe the original LPP algorithm [17] based on which we propose our OSLPP in the next subsection. LPP aims at learning a favourable low-dimensional subspace where the local structures of data in the original feature space can be well preserved. To reduce the computational complexity, an unsupervised dimensionality reduction algorithm PCA [35] is first applied to the features. Suppose $x_i \in \mathbb{R}^{d_0}$ and $x_j \in \mathbb{R}^{d_0}$ are two data points in the PCA-reduced feature space, LPP aims at learning a projection matrix $P \in \mathbb{R}^{d_0 \times d}$ ($d << d_0$) so that data points close to each other in the original space will still be close in the projected subspace. The objective of LPP [17], [16] can be formulated as:

$$\min_P \sum_{i,j} ||P^T x_i - P^T x_j||_2^2 W_{ij}, \qquad (1)$$

where W is the similarity matrix of the graph constructed by all the data points. According to [17], the edges of the graph can be created by either ϵ−neighbourhoods or k-nearest neighbours. The edge weights can be determined by the heat kernel $W_{ij} = e^{-\frac{||x_i - x_j||^2}{t}}$ or the simple binary assignment (i.e. all edges have the weights of 1). Note that LPP is an unsupervised learning method without the need for labelling information. In the following subsection, we will describe how

to extend the LPP algorithm to solve the OSDA problems where there exist unknown classes in the target domain.

C. Open-Set LPP

Open set LPP aims at exploring the structural information underlying the labelled source data and the pseudo-labelled target data including those pseudo-labelled as known classes and unknown classes. We denote the combined labelled source and pseudo-labelled target samples as $X = \{x_1^s, ..., x_{n_s}^s, x_1^t, ..., x_{\hat{n}_t}^t\} \in \mathbb{R}^{d_0 \times (n_s + \hat{n}_t)}$, where $\hat{n}_t$ is the number of pseudo-labelled target samples. To simplify the notation, we omit the superscript and assign the subscripts from 1 to $n_s + \hat{n}_t$ for the combined source and target samples, then the objective becomes the same as (1) but with a different way of constructing the similarity matrix $W \in \mathbb{R}^{(n_s + \hat{n}_t) \times (n_s + \hat{n}_t)}$:

$$\mathbf{W}_{ij} = \begin{cases} 1, & y_i = y_j \\ 0, & otherwise. \end{cases} \qquad (2)$$

where y_i denotes either the ground-truth label of x^s from the source domain, or the pseudo-label of x^t from the target domain. It is noteworthy that the pseudo-labels in (2) can be one of the known classes or the unified unknown class. Before the last iteration when all target samples are confidently recognised, only the confidently selected or rejected samples will be considered in (2) and the rest of the target samples will be treated as uncertain samples and have 0 similarity with all the other samples.

By optimising the objective (1) with the similarity matrix defined as (2), the samples labelled or pseudo-labelled as the same class will be projected to be close to each other regardless of which domain they are from. The samples from different classes will be implicitly separated in the learned subspace. Target samples rejected as unknown classes are treated as one unified class and hence will be pushed far away from all known classes implicitly. Adding terms to the objective explicitly pushing the rejected samples far away from known classes does not make a difference in our empirical

study, e.g.,

$$\min_{\boldsymbol{P}} \sum_{i,j}(||\boldsymbol{P}^T\boldsymbol{x}_i - \boldsymbol{P}^T\boldsymbol{x}_j||_2^2\boldsymbol{W}_{ij} - ||\boldsymbol{P}^T\boldsymbol{x}_i - \boldsymbol{P}^T\boldsymbol{x}_j||_2^2\boldsymbol{H}_{ij}), \quad (3)$$

where the element $\boldsymbol{H}_{ij} = 1$ if $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are from known and unknown classes, respectively and $\boldsymbol{H}_{ij} = 0$ otherwise. This variation will not be further discussed in this paper.

According to [17], optimising the objective (1) is equivalent to solving the generalised eigenvalue problem:

$$\mathbf{XDX}^T\boldsymbol{p} = \lambda(\mathbf{XLX}^T + \mathbf{I})\boldsymbol{p}, \quad (4)$$

where $\mathbf{L} = \mathbf{D} - \mathbf{W}$ is the laplacian matrix, $\mathbf{D}$ is a diagonal matrix with $\mathbf{D}_{ii} = \sum_j \mathbf{W}_{ij}$ and the regularization term $tr(\mathbf{P}^T\mathbf{P})$ is added for penalizing extreme values in the projection matrix $\mathbf{P}$. Solving the generalized eigenvalue problem gives the optimal solution $\mathbf{P} = [\boldsymbol{p}_1, ..., \boldsymbol{p}_d]$ where $\boldsymbol{p}_1, ..., \boldsymbol{p}_d$ are the eigenvectors corresponding to the largest d eigenvalues.

In summary, the key of OSLPP is to treat the target samples differently when constructing the similarity matrix $\mathbf{W}$ based on whether they are selected, rejected or uncertain. In the following subsection, we will describe how to make such decisions for the target samples.

D. Pseudo-Labelled Sample Selection and Rejection

Once the projection matrix $\boldsymbol{P}$ is learned, all the samples can be projected into the common subspace by:

$$\boldsymbol{z}^{s/t} = \boldsymbol{P}^T\boldsymbol{x}^{s/t} \quad (5)$$

To enhance the separability of different classes, we apply zero-mean centering and $l2$ normalization to each projection $\boldsymbol{z}$ so that it will be distributed on the surface of the unit hypersphere. The pseudo labelling can be done in the subspace by the *Nearest Class Mean (NCM)* method [16]. Alternatively, one can train a parametric model (e.g., Support Vector Machines, Neural Networks, etc.) for pseudo-labelling. Our choice of NCM is inspired by its simplicity and effectiveness validated in prior works [16], [36], [23]. The class means are computed over all the labelled source samples and selected pseudo-labelled target samples:

$$\bar{\boldsymbol{z}}_c = \frac{1}{n_c^s + \hat{n}_c^t}(\sum_{y_i^s = c}\boldsymbol{z}_i^s + \sum_{\hat{y}_i^t = c}\boldsymbol{z}_i^t) \quad (6)$$

and the pseudo label for a given target sample $\boldsymbol{x}^t$ can be predicted as:

$$\hat{y}^t = \arg\min_c dist(\boldsymbol{z}^t, \bar{\boldsymbol{z}}_c) \quad (7)$$

with the probability:

$$p(\hat{y}^t = c) = \frac{e^{-dist(\boldsymbol{z}^t, \bar{\boldsymbol{z}}_c)}}{\sum_c e^{-dist(\boldsymbol{z}^t, \bar{\boldsymbol{z}}_c)}} \quad (8)$$

where $dist(\boldsymbol{a}, \boldsymbol{b})$ is the Euclidean distance between $\boldsymbol{a}$ and $\boldsymbol{b}$.

In the k-th ($k = 0, 1, ..., T$) iteration of learning, we select the top $(k+1)/T$ of the target samples pseudo-labelled as c-th class for each class $c = 1, 2, ..., |C|$. These selected samples together with their corresponding pseudo-labels will be used in the next iteration of subspace learning. To reject target samples as unknown classes, we use the 1-Nearest-Neighbour

---

Algorithm 1. Open-Set LPP.

Input: Labelled source data set $\mathcal{D}^s = \{(\boldsymbol{x}_i^s, y_i^s)\}, i = 1, 2, ..., n_s$ and unlabelled target data set $\mathcal{D}^t = \{\boldsymbol{x}_i^t\}, i = 1, 2, ..., n_t$, dimensionality of the subspace d, number of iteration T, number of initial rejected target samples n_r.

Output: The projection matrix $\mathbf{P}$ and predicted labels $\{\hat{y}^t\}$ for target samples.

1: Initialize $k = 0$;
2: Learn the projection $\mathbf{P}_0$ using only source data $\mathcal{D}^s$;
3: Assign pseudo labels for all target data using Eq. (7-8);
4: Initialize n_r rejected samples of the lowest pseudo-labelling probabilities;
5: **while** $k < T$ **do**
6: $\quad k \leftarrow k + 1$;
7: $\quad$ Select a subset of pseudo-labelled target data $\mathcal{S}_k \subset \mathcal{D}^t$;
8: $\quad$ Reject samples using 1-Nearest-Neighbour and enrich the subset of rejected samples $\mathcal{R}_k \subset \mathcal{D}^t$;
9: $\quad$ Learn $\boldsymbol{P}_k$ using $\mathcal{D}^s$, $\mathcal{S}_k$ and $\mathcal{R}_k$;
10: $\quad$ Update pseudo labels for all target data using Eq.(7).
11: **end while**

---

algorithm. Any target sample will be rejected if its nearest neighbour (excluding the samples neither selected nor rejected) in the subspace is a rejected sample. In the first iteration of learning, n_r seed samples are rejected if their pseudo-labelling probabilities are top n_r lowest, where n_r is a hyper-parameter. The set of rejected samples will be enriched in each iteration without replacement. As a result, the value of n_r will be hyper-parameter trading off the accuracy of known classes and the unified unknown class.

Specifically, in the first iteration (i.e. $k = 0$), the projection matrix $\boldsymbol{P}$ is learned with source data only. Subsequently, $1/T$ of the target samples are selected for the second iteration. In the second last iteration (i.e. $k = T - 1$), all target samples will be selected or rejected for the learning of $\boldsymbol{P}$ in the final iteration. In the final iteration, we will get our final results without the need to select or reject. We summarize the proposed approach in Algorithm 1.

E. Computational Complexity

The complexity of PCA is $O(d_0 n^2 + d_0^3)$. The complexity of OSLPP is $O(2n^2 d_{PCA} + d_{PCA}^3)$ which is repeated for T times and leads to approximately $O(T(2n^2 d_{PCA} + d_{PCA}^3))$. In our experiments, we use a laptop with an Intel Core i5-7300HQ CPU and 32G memory RAM. Running 6 tasks of the Office31 dataset takes approximately 14 seconds and running 12 tasks of the Office-Home dataset takes approximately 10 minutes. For the VisDA17 and Syn2Real-O datasets, due to the large number of samples, it takes approximately 21 minutes to finish 1 task. This is much more efficient than the methods requiring the end-to-end training of a deep model which usually takes hours on a GPU.

Regarding memory usage, when the number of samples $n = n_s + n_t$ is much greater than the dimensionality, the memory complexity is $O(n^2)$. As a result, our method has the limitation

of scaling up to extremely large datasets (e.g., $n > 100,000$) for which the neural networks-based approaches can be better choices.

V. EXPERIMENTS AND RESULTS

In this section, we demonstrate the experiments and results for validating the effectiveness of the proposed method. Specifically, we introduce the datasets, experimental settings, evaluation metrics and experimental results.

A. Datasets

Four commonly used datasets for OSDA were employed in our experiments: Office-31 and Office-Home. **Office31** [37] consists of three domains: Amazon (A), Webcam (W) and DSLR (D). There are 31 common classes for all three domains containing 4,110 images in total. Following the open set protocol employed in [8], [10], we use the first 10 classes in alphabetic order as the shared known classes in both source and target domains and the last 11 classes as the unknown classes in the target domain. Image features are extracted by the ResNet50 [38] model pre-trained on ImageNet [39] without fine-tuning on the Office31 dataset. **Office-Home** [40] consists of four different domains: Artistic images (Ar), Clipart (Cl), Product images (Pr) and Real-World images (Rw). There are 65 object classes in each domain with a total number of 15,588 images. We follow [10] and use images from the first 25 classes in alphabetical order as the shared known classes in both domains and images from the remaining 40 classes as unknown classes in the target domain. Image features are extracted by the ResNet50 [38] model pre-trained on ImageNet [39] without fine-tuning on the Office-Home dataset. **VisDA17** [41] contains 12 categories in two domains. The *Synthetic* domain consists of 152,397 synthetic images from the train set and the *Real* domain contains 55,388 real-world images from the validation set. We follow the setting in [8] using 6 categories as the known classes and the remaining 6 categories as the unknown classes. **Syn2Real-O** [42] (or VisDA18) is constructed from the VisDA17 and significantly increases the openness to 0.9 by adding additional unknown samples in the target domain. Following the official setting, the source domain contains 12 known categories from the train set and the target domain contains 12 known categories plus additional unknown data from the validation set.

B. Implementation details

We implement the proposed method in MATLAB R2020b [1]. We extract deep features from the penultimate layer of the pre-trained models without fine-tuning them on the source data. The 2048-dimensional ResNet50 features or 4096-dimensional VGG16 features are first $l2$ normalised [16] and the dimensionality is reduced by PCA[35] to 16, 512, 64 and 256 for Office31, Office-Home, VisDA17 and Syn2Real-O datasets, respectively. Applying PCA to the original high-dimensional deep features not only can reduce the computation cost of OSLPP, but also benefit the performance of our experiments.

[1]Code is available from https://github.com/hellowangqian/oslpp

The dimensionality d of the learned subspace is set as 16, 128, 64 and 256 respectively. The number of iterations is set to 10 for all datasets. The number of initially rejected samples n_r is set to 140, 1200, 200 and 14,000 respectively. The sensitivity of our method to these hyper-parameters will be discussed later in this section. To be noted, we only use 10% of the source samples for the VisDA17 and Syn2Real-O datasets to reduce memory usage by our algorithm. The final results are the average over 10 trials of experiments in which each trial uses 10% evenly sampled source samples and all the target samples.

C. Evaluation metrics

There exist several evaluation metrics for evaluating OSDA approaches [10]. **OS** is the mean per-class accuracy over all target domain images from known classes and unknown classes as one unified class. To measure the capabilities of recognizing the known and unknown classes, we use the metrics **OS*** which is the mean per-class accuracy over the shared known classes and **UNK** which is the accuracy of images from the unknown classes (as one unified class).

$$OS = \frac{1}{|C_s| + 1} \sum_{i=1}^{|C_s|+1} \frac{|x : x \in \mathcal{D}_t^i \wedge \hat{y}(x) = i|}{|x : x \in \mathcal{D}_t^i|} \quad (9)$$

$$OS^* = \frac{1}{|C_s|} \sum_{i=1}^{|C_s|} \frac{|x : x \in \mathcal{D}_t^i \wedge \hat{y}(x) = i|}{|x : x \in \mathcal{D}_t^i|} \quad (10)$$

$$UNK = \frac{|x : x \in \mathcal{D}_t^{unk} \wedge \hat{y}(x) = unk|}{|x : x \in \mathcal{D}_t^{unk}|} \quad (11)$$

where $|C_s|$ is the number of source domain classes (i.e. the number of shared known classes) in the class space C_s, $\mathcal{D}_t^i$ denotes the data set of i-th class in the target domain and $\hat{y}(x)$ is the predicted label of data sample x.

OS is a combination of OS* and UNK as $OS = \frac{|C_s|}{|C_s|+1} \times OS^* + \frac{1}{|C_s|+1} \times UNK$, however, as pointed out in [10], OS can be dominated by the accuracy of known classes since the unknown classes are treated as one unified class. One effective evaluation metric properly balancing the recognition performance for known and unknown classes is the harmonic mean of OS* and UNK:

$$HOS = \frac{2 \times OS^* \times UNK}{OS^* + UNK} \quad (12)$$

In most of our experiments, we report OS*, UNK and HOS for individual domain adaptation tasks as well as the average performance over all possible tasks for a dataset. In the ablation study, we report one extra evaluation metric **ALL** which is the instance-wise accuracy of all target samples.

$$ALL = \frac{\sum_{i=1}^{|C_s|+1} |x : x \in \mathcal{D}_t^i \wedge \hat{y}(x) = i|}{\sum_{i=1}^{|C_s|+1} |x : x \in \mathcal{D}_t^i|} \quad (13)$$

TABLE II
OPEN-SET CLASSIFICATION RESULTS (%) ON OFFICE31 DATASET USING EITHER RESNET50 FEATURES OR RESNET50 BASED DEEP MODELS († INDICATES ALEXNET WAS USED).

Method	A→D			A→W			D→A			D→W			W→A			W→D			Average		
	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS
STA$_{max}$ [9]	91.0	63.9	75.0	86.7	67.6	75.9	83.1	65.9	73.2	94.1	55.5	69.8	66.2	68.0	66.1	84.9	67.8	75.2	84.3	64.8	72.6
OSBP [8]	90.5	75.5	82.4	86.8	79.2	82.7	76.1	72.3	75.1	97.7	96.7	**97.2**	73.0	74.4	73.7	99.1	84.2	91.1	87.2	80.4	83.7
UAN [26]	95.6	24.4	38.9	95.5	31.0	46.8	93.5	53.4	68.0	99.8	52.5	68.8	94.1	38.8	54.9	81.5	41.4	53.0	93.4	40.3	55.1
SE-CC† [32]	84.0	46.6	59.9	84.2	64.4	73.0	90.3	12.2	21.5	96.6	55.9	70.8	85.9	73.8	84.6	99.1	73.8	84.6	90.0	50.6	62.3
ROS [10]	87.5	77.8	82.4	88.4	76.7	82.1	74.8	81.2	77.9	99.3	93.0	96.0	69.7	86.6	77.2	100.0	99.4	**99.7**	86.6	85.8	85.9
OSLPP (Ours)	92.6	90.4	**91.5**	89.6	88.8	**89.0**	82.2	77.1	**79.3**	96.9	88.4	92.3	79.0	79.3	**78.7**	95.8	91.5	93.6	89.3	85.9	**87.6**

TABLE III
OPEN-SET CLASSIFICATION RESULTS (%) ON OFFICE-HOME DATASET USING EITHER RESNET50 FEATURES OR RESNET50 BASED DEEP MODELS.

Method	Ar→Cl			Ar→Pr			Ar→Rw			Cl→Ar			Cl→Pr			Cl→Rw		
	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS
STA$_{max}$ [9]	46.0	72.3	55.8	68.0	48.4	54.0	78.6	60.4	68.3	51.4	65.0	57.4	61.8	59.1	60.4	67.0	66.7	66.8
OSBP [8]	50.2	61.1	55.1	71.8	59.8	65.2	79.3	67.5	72.9	59.4	70.3	64.3	67.0	62.7	64.7	72.0	69.2	70.6
UAN [26]	62.4	0.0	0.0	81.1	0.0	0.0	88.2	0.1	0.2	70.5	0.0	0.0	74.0	0.1	0.2	80.6	0.1	0.2
ROS [10]	50.6	74.1	60.1	68.4	70.3	69.3	75.8	77.2	76.5	53.6	65.5	58.9	59.8	71.6	65.2	65.3	72.2	68.6
DAOD [12]	72.6	51.8	60.5	55.3	57.9	56.6	78.2	62.6	69.5	59.1	61.7	60.4	70.8	52.6	60.4	77.8	57.0	65.8
PGL [11]	63.3	19.1	29.3	78.9	32.1	45.6	87.7	40.9	55.8	85.9	5.3	10.0	73.9	24.5	36.8	70.2	33.8	45.6
OSLPP (Ours)	55.9	67.1	**61.0**	72.5	73.1	**72.8**	80.1	69.4	74.3	49.6	79.0	60.9	61.6	73.3	**66.9**	67.2	73.9	70.4

Method	Pr→Ar			Pr→Cl			Pr→Rw			Rw→Ar			Rw→Cl			Rw→Pr			Average		
	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS
STA$_{max}$ [9]	54.2	72.4	61.9	44.2	67.1	53.2	76.2	64.3	69.5	67.5	66.7	67.1	49.9	61.1	54.5	77.1	55.4	64.5	61.8	63.3	61.1
OSBP [8]	59.1	68.1	63.2	44.5	66.3	53.2	76.2	71.7	73.9	66.1	67.3	66.7	48.0	63.0	54.5	76.3	68.6	72.3	64.1	66.3	64.7
UAN [26]	73.7	0.0	0.0	59.1	0.0	0.0	84.0	0.1	0.2	77.5	0.1	0.2	66.2	0.0	0.0	85.0	0.1	0.1	75.2	0.0	0.1
ROS [10]	57.3	64.3	60.6	46.5	71.2	56.3	70.8	78.4	**74.4**	67.0	70.8	**68.8**	51.5	73.0	60.4	72.0	80.0	**75.7**	61.6	72.4	66.2
DAOD [12]	71.3	50.5	59.1	58.4	42.8	49.4	81.8	50.6	62.5	66.7	43.3	52.5	60.0	36.6	45.5	84.1	34.7	49.1	69.6	50.2	57.6
PGL [11]	73.7	34.7	47.2	59.2	38.4	46.6	84.8	27.6	41.6	81.5	6.1	11.4	68.8	0.0	0.0	84.8	38.0	52.5	76.1	25.0	35.2
OSLPP (Ours)	54.6	76.2	**63.6**	53.1	67.1	**59.3**	77.0	71.2	74.0	60.8	75.0	67.2	54.4	64.3	59.0	78.4	70.8	74.4	63.8	71.7	**67.0**

TABLE IV
OPEN-SET CLASSIFICATION RESULTS (%) ON THE VISDA17 DATASET USING VGG16 AND THE SYN2REAL-O DATASET USING RESNET50.

Method	VisDA17				Syn2Real-O			
	OS*	UNK	OS	HOS	OS*	UNK	OS	HOS
DANN [43]+OSVM	57.8	41.9	55.5	48.6	47.4	33.6	46.3	39.3
OSBP [8]	59.2	85.1	62.9	69.8	47.7	79.3	50.1	58.9
STA [9]	63.9	84.1	66.8	72.6	52.2	59.1	52.7	55.4
InheritTune [33]	64.7	88.5	68.1	74.8	-	-	-	-
PGL [11]	82.8	68.6	80.7	75.0	66.8	49.6	65.5	56.9
OSLPP (Ours)	68.6	88.2	71.4	**76.1**	61.7	70.0	62.4	**65.6**

D. Comparison with state of the art

The proposed method OSLPP is compared against recent state-of-the-art approaches to OSDA problems. The approaches we compare against include STA [9], OSBP [8], UAN [26], ROS [10], DAOD [12] and PGL [11]. We do not compare with [44], [33] and [32] since they either use a different experimental protocol or different deep features in their experiments and make it difficult for a direct comparison with those considered in our work.

To make a fair and reliable comparison, we report the results of state-of-the-art methods reproduced by [10] for STA, OSBP, UAN and ROS. For DAOD and PGL, the metrics UNK and HOS are calculated by Eqs. (9-12) based on the reported OS and OS* values in their original papers.

As we can see in Table II, our proposed method achieves the best average HOS of 87.6% followed by 85.9% by [10] and 83.7% by [8]. Among the six adaptation tasks, our method performs the best in terms of HOS on four of them. The other methods usually perform well in recognising the known classes (high OS*) but are bad at recognising the unknown classes (low UNK). This is partially due to the bias caused by only considering the evaluation metrics of OS and OS*. A high OS* along with a low UNK means the method mistakenly classifies a large number of samples from unknown classes as one of the known classes. This caveat cannot be disclosed when using only OS* and OS as the evaluation metrics.

On the more challenging Office-Home dataset, as shown in Table III, our method achieves the best or the second best HOS on all 12 tasks and the best average HOS of 67.0% followed by 66.2% by [10] and 64.7% by [8]. The experimental results on both datasets are consistent in that methods achieving high OS* do not necessarily perform well in practice since they may make too many mistakes on the target samples from unknown classes as we can see from the results of UAN [26] and PGL [11].

On the VisDA17 and Syn2Real-O datasets, our method also achieves the best performance in terms of HOS as shown in Table IV. PGL [11] performs the best in recognising known classes but suffers a lower UNK. This drawback becomes obvious when the openness is high as in the Syn2Real-O dataset for which the overall recognition accuracy of PGL is 51.5% [11] whilst our method achieves 69.4%.

E. Comparison with universal domain adaptation approaches

We compare our method with universal domain adaptation approaches on the open-set domain adaptation setting. The universal domain adaptation approaches are usually more powerful with more complex modules designed and integrated. The results of HOS on three datasets are shown in Table V

Method	Office31	OfficeHome	VisDA17	Average
DCC [29]	72.6	61.7	59.6	64.6
DANCE [27]	79.8	63.0	67.5	70.1
OvaNet [30]	**91.7**	64.0	66.1	73.9
GATE [45]	<u>89.5</u>	**69.1**	<u>70.8</u>	76.5
OSLPP(Ours)	87.6	<u>67.0</u>	**76.1**	**76.9**

from which we can see our proposed OSLPP can achieve comparable or superior performance to most universal domain adaptation approaches. Although none of the investigated methods including ours performs the best on all three datasets, our proposed method achieves the best average performance with on HOS of 76.9%. In addition, our approach is more computationally efficient as it works on top of deep features in contrast to GATE [45] and OvaNet [30] both of which involve training a deep CNN model.

F. Effect of Hyper-Parameters

We investigate the effect of four hyper-parameters in our method: the dimensionality of the PCA subspace d_{PCA}, the dimensionality of the OSLPP subspace d, the number of initially rejected samples n_r and the number of iterations T. To these ends, we set the investigated hyper-parameter to the values within a pre-defined set and the others fixed as the default in our experiments. The experiments are conducted on the Office31 and Office-Home datasets. The investigated values for four hyper-parameters are $d_{PCA} \in \{1024, 512, 256, 128, 64, 32, 16, 8\}$, $d \in \{512, 256, 128, 64, 32, 16, 8\}$, $n_r \in \{40, 60, 80, 100, 120, 140, 160, 180, 200, 220\}$ ($\times 10$ for the Office-Home dataset since this dataset contains much more samples) and $T \in \{6, 8, 10, 12, 14, 16, 18, 20\}$, respectively.

The results are shown in Figure 3. We report the average OS*, UNK and HOS for each experiment. The optimal number of PCA dimensionality for Office31 is 16 and a smaller or greater value will lead to a slight performance drop for both OS* and UNK. This result means the ResNet50 features are discriminative enough to separate 10 shared known classes in this simple dataset so that the unsupervised PCA can extract the most useful information in the first 16 principal components whilst more dimensions hurt the performance slightly. Similarly, the optimal dimensionality of OSLPP subspace is 16 for Office31 and more dimensions lead to slightly worse performance. For the more challenging Office-Home dataset with 25 shared known classes and 40 unknown classes, the average HOS is less sensitive to the subspace dimensionality and optimal average HOS can be achieved within a large range of values for d_{PCA} (i.e. 32-1024) and d (i.e. 32-512). In addition, the values of d_{PCA} and d also affect the trade-off between the OS* and UNK although their harmonic mean HOS is marginally affected.

The effects of n_r and T are more understandable and consistent on the two datasets. On one hand, the performance of our method in terms of HOS is not sensitive to these two

hyper-parameters given that the optimal HOS can be achieved with a large range of values for n_r (i.e. 40-180 for Office31 and 400-1800 for Office-Home) and T (i.e. 8-20). On the other hand, both the n_r and T control the trade-off between the recognition accuracy of known classes OS* and the accuracy of unknown classes UNK. Increasing the number of initially rejected samples n_r will lead to more samples recognised as unknown classes hence an increased UNK. Increasing the number of iterations T means selecting samples as known classes more slowly whilst the pace of rejecting samples as unknown classes is not affected. As a result, more samples will be rejected and recognised as unknown classes after more iterations hence a higher UNK can be achieved. Along with the improvement of UNK, our method suffers from the decrease of OS* although their harmonic mean HOS is stable.

Overall, our proposed method is less sensitive to hyperparameters. In addition, we can trade off the recognition accuracy of known classes and unknown classes by adjusting the hyper-parameters of n_r and T according to the requirements in practice.

G. Ablation study

An ablation study is conducted on the VisDA17 dataset by removing different components of the proposed method. Specifically, we remove the component of PCA (denoted as "w/o PCA"), the component of OSLPP (denoted as "w/o OSLPP"), and the component of pseudo-label selection (denoted as "w/o selection"), respectively. We also replace the OSLPP algorithm with LPP (i.e. without considering the target samples rejected as unknown classes) and denote it as "w/o rejection". The experimental results are shown in Table VI. Overall, our proposed method with the OSLPP algorithm performs the best in all four benchmark datasets. Replacing the OSLPP with the conventional LPP (i.e. w/o rejection) leads to a marginal performance drop due to the lower UNK. Such a difference in performance is accounted for by the fact that OSLPP considers the unknown classes as a unified one during the discriminative subspace learning whilst LPP does not. The worst HOS is obtained by the method "w/o selection" in which the pseudo-labelled target samples are not progressively selected but over-confidently included for the subspace learning in the next iteration. As a result, the known classes are recognised with a better chance (i.e. high OS*) at the cost of recognition accuracy of unknown classes (i.e. extremely low UNK). When the OSLPP is removed from our framework (i.e. w/o OSLPP), the recognition is done in the PCA subspace and the performance is significantly worse than our proposed method with OSLPP. Similarly, the removal of PCA also causes a significant performance drop when compared with our proposed approach.

H. The effect of the number of source samples

We conduct experiments to investigate how the number of source samples affects the performance of our method. To this end, we evenly select 50%, 20%, 10%, 5%, 2% or 1% of the source samples in the experiments and compare the results against the baseline when 100% (or 10% for the VisDA17
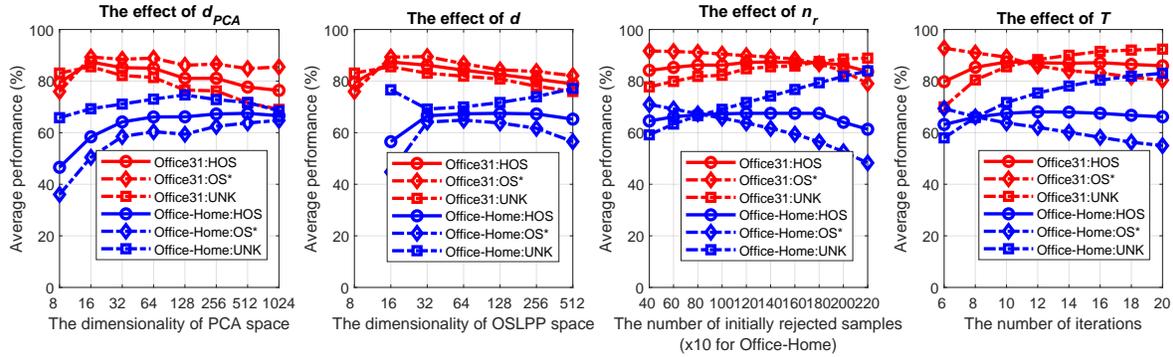
Fig. 3. The effect of hyper-parameters on the Office31 and Office-Home datasets.

TABLE VI
RESULTS OF THE ABLATION STUDY.

Method	Office31					Office-Home					VisDA17					Syn2Real-O				
	OS*	UNK	OS	ALL	HOS	OS*	UNK	OS	ALL	HOS	OS*	UNK	OS	ALL	HOS	OS*	UNK	OS	ALL	HOS
w/o PCA	86.2	69.1	84.6	76.9	76.1	66.5	65.1	66.4	65.8	65.2	61.7	69.1	62.8	62.9	65.1	55.9	69.2	56.9	**69.9**	61.8
w/o OSLPP	89.0	84.0	88.6	86.6	86.4	59.8	67.4	60.1	64.3	62.8	63.1	77.3	65.1	66.8	69.5	38.2	58.9	39.8	57.0	46.3
w/o selection	**93.1**	45.4	88.8	67.3	57.4	**93.1**	45.4	**88.8**	67.3	57.4	**73.6**	0.6	63.2	44.9	1.2	**62.8**	27.3	60.0	31.2	38.1
w/o rejection	89.8	82.2	**89.1**	85.7	85.8	64.9	70.4	65.1	68.2	**67.0**	65.2	85.2	68.1	70.2	73.9	61.6	69.6	62.2	68.5	65.4
OSLPP (Ours)	89.3	**85.9**	89.0	**87.6**	**87.6**	63.8	**71.7**	64.0	**68.5**	**67.0**	68.6	**88.2**	71.4	**73.6**	**76.1**	61.7	**70.0**	62.4	69.0	**65.6**

TABLE VII
OPEN-SET CLASSIFICATION RESULTS (%) ON OFFICE31 DATASET USING DIFFERENT NUMBERS OF SOURCE SAMPLES.

Fraction of source samples	A→D			A→W			D→A			D→W			W→A			W→D			Average		
	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS
100%	92.6	90.4	91.5	89.6	88.8	89.0	82.2	77.1	79.3	96.9	88.4	92.3	79.0	79.3	78.7	95.8	91.5	93.6	89.3	85.9	87.6
50%	92.0	85.6	88.7	89.0	84.6	86.8	82.7	75.8	79.1	96.6	86.9	91.5	82.8	71.6	76.8	95.4	95.7	95.6	89.7	83.4	86.4
20%	88.6	85.1	86.8	90.2	91.0	90.6	80.1	68.8	74.0	93.1	86.5	89.7	83.9	73.8	78.5	92.7	91.0	91.8	88.1	82.7	85.2
10%	89.8	85.6	87.7	85.7	84.6	85.2	60.7	78.0	68.3	60.0	92.1	72.7	73.5	65.6	69.3	82.3	91.0	86.4	75.3	82.8	78.3

TABLE VIII
OPEN-SET CLASSIFICATION RESULTS (%) ON OFFICE-HOME DATASET USING DIFFERENT NUMBERS OF SOURCE SAMPLES.

Fraction of Source samples	Ar→Cl			Ar→Pr			Ar→Rw			Cl→Ar			Cl→Pr			Cl→Rw		
	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS
100%	55.9	67.1	61.0	72.5	73.1	72.8	80.1	69.4	74.3	49.6	79.0	60.9	61.6	73.3	66.9	67.2	73.9	70.4
50%	57.4	66.0	61.4	69.8	72.2	71.0	77.2	65.5	70.9	51.5	78.4	62.1	64.8	73.8	69.0	70.0	72.3	71.1
20%	56.1	66.5	60.8	69.4	69.6	69.5	74.8	71.2	73.0	51.7	73.5	60.7	59.9	70.8	64.9	72.1	71.9	72.0
10%	51.9	69.5	59.4	65.6	74.2	69.6	71.5	66.6	69.0	49.9	72.6	59.2	57.1	66.1	61.2	68.0	63.4	65.6
5%	38.2	69.0	49.2	49.8	66.3	56.9	55.0	69.1	61.2	48.1	71.7	57.6	56.7	65.2	60.7	64.0	61.0	62.5
2%	21.5	73.5	33.3	27.0	67.9	38.7	28.3	71.0	40.5	20.8	74.7	32.5	25.5	65.0	36.6	29.2	66.6	40.6

Fraction of Source samples	Pr→Ar			Pr→Cl			Pr→Rw			Rw→Ar			Rw→Cl			Rw→Pr			Average		
	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS
100%	54.6	76.2	63.6	53.1	67.1	59.3	77.0	71.2	74.0	60.8	75.0	67.2	54.4	64.3	59.0	78.4	70.8	74.4	63.8	71.7	67.0
50%	52.8	76.2	62.4	54.1	66.2	59.6	74.9	67.0	70.7	60.6	74.1	66.7	57.0	65.9	61.1	77.3	71.7	74.4	64.0	70.8	66.7
20%	56.1	75.5	64.4	55.1	65.2	59.7	77.1	72.0	74.5	59.0	72.9	65.2	55.9	63.5	59.5	76.2	69.5	72.7	63.6	70.2	66.4
10%	54.7	74.4	63.1	55.3	62.3	58.6	76.9	64.8	70.4	59.8	74.5	66.3	55.9	63.8	59.6	71.2	67.8	69.4	61.5	68.3	64.3
5%	54.3	73.8	62.6	54.7	60.6	57.5	75.4	65.4	70.0	57.3	74.9	64.9	54.0	61.3	57.4	69.9	67.9	68.9	56.4	67.2	60.8
2%	30.6	74.4	43.4	24.3	64.8	35.3	40.5	67.0	50.5	37.4	78.6	50.7	34.3	69.5	46.0	41.7	69.6	52.2	30.1	70.2	41.7

and Syn2Real-O datasets) of the source samples are used. The experimental results on four datasets are shown in Tables VII-IX.

For the Office31 dataset, a significant performance drop can be observed when the fraction of source data decreases to 10%. The performance drop is mainly caused by the adaptation tasks where the domain D or the domain W servers as the source domain. Note that the total numbers of samples in these two domains (i.e. D and W) are 498 and 795, respectively. A fraction of 10% samples means there could exist some classes with zero samples in the source data. Similar phenomena can

be seen in Table VIII. It can be concluded that our proposed method can achieve comparably good performance even if only a very small fraction (e.g., 10%) of source samples are used.

For the VisDA17 dataset, using less than 10% source-domain data surprisingly gives better results as shown in Table IX due to the better recognition accuracy of unknown classes. On the other hand, using less than 10% source-domain data gives consistently worse performance for the Syn2Real-O dataset due to the decreasing accuracy of recognising known classes (i.e., OS*).

Fraction of	VisDA17				Syn2Real-O			
source samples	OS*	UNK	OS	HOS	OS*	UNK	OS	HOS
10%	68.6	88.2	71.4	76.1	61.7	70.0	62.4	65.6
5%	69.5	92.3	72.8	79.3	59.3	69.4	60.1	63.9
2%	69.9	93.6	73.3	80.0	54.1	69.6	55.3	60.8
1%	70.0	92.7	73.3	79.8	49.3	69.4	50.8	57.4

(a) Original feature space	(b) Learned subspace by LDA	(c) Learned subspace by OSLPP

Fig. 4. Feature visualisation via t-SNE. Source and target samples are represented by 'o' and '+' symbols respectively; different known classes are denoted by colours and the unknown classes are denoted by black '×'.

I. Openness analysis

We follow [10] to make an empirical openness analysis on the Office-Home dataset. Specifically, we set 40 (class IDs: 1-40, 16-55, 26-65), 25 (class IDs: 1-25, 26-50, 41-65), 10 (class IDs: 1-10, 11-20, 21-30) and 5 (class IDs: 1-5, 6-10, 11-15) classes as known classes, respectively. The class IDs from 1 to 65 are defined by ranking all the classes in alphabetical order.

The results are shown in Table X and the results of three comparative methods are from [10]. Our proposed method OSLPP outperforms others in settings where there are 40/25 known classes and is comparable to ROS [10] when there are 10/5 known classes. These results indicate our method is more advantageous when the openness is moderate.

J. Extend to transductive open-set recognition

To investigate how our proposed approach works for the transductive open-set recognition (OSR) problems without domain shift, we design and conduct a series of experiments. On the Office-Home dataset, we divide each domain into two disjoint subsets: one for training and the other for testing. (a) We simulate the open-set recognition problem under the transductive setting. The training data and the test data are from the same domain but the test data contain additional classes which are not present in the training data. (b) We simulate the open-set domain adaptation problem. The training data and the test data are from the source and target domains respectively. For both (a) and (b), we apply our full method and the ablated one without the OSLPP algorithm for comparison. The results are shown in Table XI. Based on the experimental results, following conclusions can be drawn: (i) our proposed approach can be directly applied to the open-set recognition problems (i.e. the tasks of Pr→Pr and Rw→Rw) under the transductive setting; (ii) for the transductive OSR, the high performance of our approach is mainly contributed by the progressive sample selection and rejection strategy, given that removing the OSLPP algorithms only degrades the performance marginally; (iii) for OSDA, the OSLPP algorithm plays a more crucial role in bridging the domain gap, thereby significant performance drops are observed when it is not used.

In summary, the experimental results in Table XI demonstrate our proposed method also works for OSR under the transductive setting, and it is the OSLPP algorithm that enables our proposed approach to bridge the domain gap when it does exist.
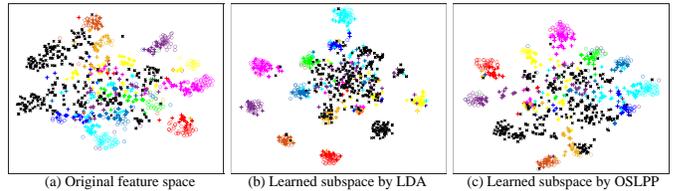
K. Visual inspection

We take the task Art→Clipart as an example to inspect how the class separability can be improved in the learned subspace by visualising the features with the t-SNE technique. To avoid clutters, we select the first 10 shared known classes and the first 10 unknown classes for visualisation. As shown in Figure 4, the samples from the source domain ('o') and the target domain ('+') in the learned common subspace are better aligned class-wise whilst the unknown classes ('×') are also better separated from the known classes.

Other than the original features (a) and the features learned by our proposed OSLPP (c), we also visualize the features learned by LDA (b) as a representative of existing methods which push unknown classes into a compacted region in the learned space as illustrated in Figure 1 and demonstrated in Figure 4. As a result, our proposed OSLPP algorithm outperforms LDA in the domain adaptation tasks (e.g., the HOS values are 67.0% and 65.0% respectively on the Office-Home dataset).

VI. DISCUSSION AND CONCLUSION

We address the OSDA problem in the image classification domain by proposing a novel OSLPP algorithm and a progressive pseudo-labelled sample selection and rejection algorithm. The OSLPP adapts the original LPP algorithm to the OSDA scenario by considering the labelled source samples and pseudo-labelled target samples which have been either selected or rejected. Experimental results on two benchmark datasets demonstrate our proposed method can perform comparably or outperform state-of-the-art approaches to OSDA and is more efficient. The method is also less sensitive to the hyper-parameters in terms of the harmonic mean but provides the flexibility of trading off the accuracy of known and unknown classes which can be useful in real-world applications.

Our method suffers from the common issue of how to set proper hyper-parameters to adjust the recognition accuracy of known classes and unknown classes. In a real-world scenario, we do not have prior knowledge of how many unlabelled target samples are from unknown classes if there are any. In our method, this issue corresponds to the question of how to set the values of n_r and T given a domain adaptation task to achieve the best performance. This is also the key problem in Universal Domain Adaptation problems [26] and will be left to future work.

TABLE X
OPENNESS ANALYSIS ON THE OFFICE-HOME DATASET.

Method	40 known classes				25 known classes			
	OS*	UNK	OS	HOS	OS*	UNK	OS	HOS
STAsum [9]	53.5±1.7	71.1±3.0	53.9±1.7	60.1±1.7	63.3±3.3	65.1±2.2	63.5±2.6	61.4±1.5
OSBP [8]	55.9±1.6	71.8±3.6	56.3±1.6	62.3±1.8	61.0±2.8	64.2±2.8	61.4±2.5	62.0±2.8
ROS [10]	55.9±1.2	71.1±2.8	56.2±1.2	62.1±1.8	59.5±1.9	70.4±4.0	59.9±1.9	63.9±2.7
OSLPP (Ours)	61.9±1.0	72.4±4.9	62.1±1.1	**66.3±2.7**	63.7±2.0	74.1±2.5	64.1±2.0	**68.0±2.0**
	10 known classes				5 known classes			
	OS*	UNK	OS	HOS	OS*	UNK	OS	HOS
STAsum [9]	73.8±6.6	61.4±14.5	72.5±5.8	62.5±8.6	78.9±6.3	52.1±17.7	74.4±2.8	57.5±12.3
OSBP [8]	76.2±5.7	47.9±4.8	73.6±5.7	58.0±5.1	82.5±5.6	21.5±1.6	72.3±4.9	33.3±2.6
ROS [10]	68.9±5.5	78.9±2.9	69.9±5.2	**73.0±4.4**	71.1±7.6	86.3±2.7	73.4±6.7	**77.2±5.9**
OSLPP (Ours)	74.3±5.2	74.3±2.6	74.3±4.8	**73.6±3.1**	73.1±5.6	76.8±0.4	73.7±4.7	**73.6±3.4**

TABLE XI
COMPARISON OF OPEN-SET RECOGNITION PERFORMANCE (HOS) WITH
AND WITHOUT THE DOMAIN SHIFT.

target domain →	Pr				Rw			
source domain →	Ar	Cl	Rw	**Pr**	Ar	Cl	Pr	**Rw**
Ours w/o OSLPP	64.8	60.4	67.7	**78.7**	71.9	68.3	70.0	**75.0**
Ours	69.5	65.3	70.1	**78.8**	72.6	70.7	72.0	**75.8**
Gain	4.7	4.9	2.4	**0.1**	0.7	2.4	2.0	**0.8**

REFERENCES

[1] Q. Wang and T. P. Breckon, "Generalized zero-shot domain adaptation via coupled conditional variational autoencoders," *Neural Networks*, vol. 163, pp. 40–52, 2023.

[2] Z. Yi, W. Shang, T. Xu, and X. Wu, "Neighborhood preserving and weighted subspace learning method for drift compensation in gas sensor," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 52, no. 6, pp. 3530–3541, 2021.

[3] J. Tian, J. Zhang, W. Li, and D. Xu, "Vdm-da: Virtual domain modeling for source data-free domain adaptation," *IEEE Transactions on Circuits and Systems for Video Technology*, 2021.

[4] X. Xu, H. He, H. Zhang, Y. Xu, and S. He, "Unsupervised domain adaptation via importance sampling," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 12, pp. 4688–4699, 2019.

[5] C. Chen, W. Xie, W. Huang, Y. Rong, X. Ding, Y. Huang, T. Xu, and J. Huang, "Progressive feature alignment for unsupervised domain adaptation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 627–636.

[6] C. Chen, Z. Chen, B. Jiang, and X. Jin, "Joint domain alignment and discriminative feature learning for unsupervised deep domain adaptation," in *AAAI Conference on Artificial Intelligence*, 2019.

[7] M. Meng, Z. Wu, T. Liang, J. Yu, and J. Wu, "Exploring fine-grained cluster structure knowledge for unsupervised domain adaptation," *IEEE Transactions on Circuits and Systems for Video Technology*, 2022.

[8] K. Saito, S. Yamamoto, Y. Ushiku, and T. Harada, "Open set domain adaptation by backpropagation," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 153–168.

[9] H. Liu, Z. Cao, M. Long, J. Wang, and Q. Yang, "Separate to adapt: Open set domain adaptation via progressive separation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2927–2936.

[10] S. Bucci, M. R. Loghmani, and T. Tommasi, "On the effectiveness of image rotation for open set domain adaptation," in *European Conference on Computer Vision*. Springer, 2020, pp. 422–438.

[11] Y. Luo, Z. Wang, Z. Huang, and M. Baktashmotlagh, "Progressive graph learning for open-set domain adaptation," in *International Conference on Machine Learning*. PMLR, 2020, pp. 6468–6478.

[12] Z. Fang, J. Lu, F. Liu, J. Xuan, and G. Zhang, "Open set domain adaptation: Theoretical bound and algorithm," *IEEE transactions on neural networks and learning systems*, 2020.

[13] J. Liu, M. Jing, J. Li, K. Lu, and H. T. Shen, "Open set domain adaptation via joint alignment and category separation," *IEEE Transactions on Neural Networks and Learning Systems*, 2021.

[14] M. Jing, J. Li, L. Zhu, Z. Ding, K. Lu, and Y. Yang, "Balanced open set domain adaptation via centroid alignment," in *Proceedings of the*

[15] Z. Feng, C. Xu, and D. Tao, "Open-set hypothesis transfer with semantic consistency," *IEEE Transactions on Image Processing*, vol. 30, pp. 6473–6484, 2021.

[16] Q. Wang and T. P. Breckon, "Unsupervised domain adaptation via structured prediction based selective pseudo-labeling," in *AAAI Conference on Artificial Intelligence*, 2020.

[17] X. He and P. Niyogi, "Locality preserving projections," in *Advances in neural information processing systems*, 2004, pp. 153–160.

[18] J. Sun, Z. Wang, W. Wang, H. Li, F. Sun, and Z. Ding, "Joint adaptive dual graph and feature selection for domain adaptation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 3, pp. 1453–1466, 2021.

[19] F. Liu, X. Zhang, F. Wan, X. Ji, and Q. Ye, "Domain contrast for domain adaptive object detection," *IEEE Transactions on Circuits and Systems for Video Technology*, 2021.

[20] Y. Tian and S. Zhu, "Partial domain adaptation on semantic segmentation," *IEEE Transactions on Circuits and Systems for Video Technology*, 2021.

[21] S. Li, M. Yuan, J. Chen, and Z. Hu, "Adadc: Adaptive deep clustering for unsupervised domain adaptation in person re-identification," *IEEE Transactions on Circuits and Systems for Video Technology*, 2021.

[22] Y. Yang, X. Yang, T. Sakamoto, F. Fioranelli, B. Li, and Y. Lang, "Unsupervised domain adaptation for disguised-gait-based person identification on micro-doppler signatures," *IEEE Transactions on Circuits and Systems for Video Technology*, 2022.

[23] Q. Wang and T. P. Breckon, "Source class selection with label propagation for partial domain adaptation," in *2021 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2021, pp. 769–773.

[24] Y. Kim and S. Hong, "Adaptive graph adversarial networks for partial domain adaptation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 1, pp. 172–182, 2021.

[25] S. Li, K. Gong, B. Xie, C. H. Liu, W. Cao, and S. Tian, "Critical classes and samples discovering for partial domain adaptation," *IEEE Transactions on Cybernetics*, 2022.

[26] K. You, M. Long, Z. Cao, J. Wang, and M. I. Jordan, "Universal domain adaptation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 2720–2729.

[27] K. Saito, D. Kim, S. Sclaroff, and K. Saenko, "Universal domain adaptation through self supervision," *Advances in Neural Information Processing Systems*, vol. 33, 2020.

[28] B. Fu, Z. Cao, M. Long, and J. Wang, "Learning to detect open classes for universal domain adaptation," in *European Conference on Computer Vision*. Springer, 2020, pp. 567–583.

[29] G. Li, G. Kang, Y. Zhu, Y. Wei, and Y. Yang, "Domain consensus clustering for universal domain adaptation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 9757–9766.

[30] K. Saito and K. Saenko, "Ovanet: One-vs-all network for universal domain adaptation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 9000–9009.

[31] Y. Zhao, Z. Zhong, Z. Luo, G. H. Lee, and N. Sebe, "Source-free open compound domain adaptation in semantic segmentation," *IEEE Transactions on Circuits and Systems for Video Technology*, 2022.

[32] Y. Pan, T. Yao, Y. Li, C.-W. Ngo, and T. Mei, "Exploring category-agnostic clusters for open-set domain adaptation," in *Proceedings of the*

IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 13 867–13 875.

[33] J. N. Kundu, N. Venkat, A. Revanur, R. V. Babu *et al.*, "Towards inheritable models for open-set domain adaptation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 12 376–12 385.

[34] Q. Wang, P. Bu, and T. P. Breckon, "Unifying unsupervised domain adaptation and zero-shot visual recognition." in *International Joint Conference on Neural Networks*, 2019.

[35] S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis," *Chemometrics and intelligent laboratory systems*, vol. 2, no. 1-3, pp. 37–52, 1987.

[36] J. Dong, Z. Fang, A. Liu, G. Sun, and T. Liu, "Confident anchor-induced multi-source free domain adaptation," *Advances in Neural Information Processing Systems*, vol. 34, pp. 2848–2860, 2021.

[37] K. Saenko, B. Kulis, M. Fritz, and T. Darrell, "Adapting visual category models to new domains," in *European Conference on Computer Vision*. Springer, 2010, pp. 213–226.

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

[39] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *IEEE conference on computer vision and pattern recognition*. IEEE, 2009, pp. 248–255.

[40] H. Venkateswara, J. Eusebio, S. Chakraborty, and S. Panchanathan, "Deep hashing network for unsupervised domain adaptation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 5018–5027.

[41] D. Bashkirova, D. Hendrycks, D. Kim, S. Mishra, K. Saenko, K. Saito, P. Teterwak, and B. Usman, "Visda-2021 competition universal domain adaptation to improve performance on out-of-distribution data," 2021.

[42] X. Peng, B. Usman, K. Saito, N. Kaushik, J. Hoffman, and K. Saenko, "Syn2real: A new benchmark for synthetic-to-real visual domain adaptation," *CoRR*, vol. abs/1806.09755, 2018.

[43] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, "Domain-adversarial training of neural networks," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 2096–2030, 2016.

[44] P. Panareda Busto and J. Gall, "Open set domain adaptation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 754–763.

[45] L. Chen, Y. Lou, J. He, T. Bai, and M. Deng, "Geometric anchor correspondence mining with uncertainty modeling for universal domain adaptation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 16 134–16 143.

Qian Wang is a visiting scholar with the Department of Computer Science at Durham University. He was a deep learning researcher at MediaTek Research Cambridge, and a research associate with the Department of Computer Science at Durham University, United Kingdom. His research focuses on deep learning and computer vision. He received his PhD in machine learning from The University of Manchester in 2017, a Master's degree in Biomedical Engineering and a BSc in Electronic Engineering in 2013 and 2010 respectively, both from the University of Science and Technology of China (Hefei).

Fanlin Meng (Senior Member, IEEE) received the B.Sc. degree in automation from the China University of Mining and Technology, Xuzhou, China, in 2008, the M.Sc. degree in systems engineering from Xiamen University, Xiamen, China, in 2011, and the Ph.D. degree in computer science from the University of Manchester, Manchester, U.K., in 2015. He is currently a Lecturer in Data Science with the University of Manchester. He is a Fellow of British Computer Society and a member of EPSRC Peer Review College. His primary research interests include energy market, smart energy and mobility, operations research, machine learning, and game theory and optimization.

Toby P. Breckon is currently a Professor within the Departments of Engineering and Computer Science, Durham University (UK). His key research interests lie in the domain of computer vision and image processing and he leads a range of research activity in this area.

Prof. Breckon holds a PhD in informatics (computer vision) from the University of Edinburgh (UK). He has been a visiting member of faculty at the Ecole Supérieure des Technologies Industrielles Avancées (France), Northwestern Polytechnical University (China), Shanghai Jiao Tong University (China) and Waseda University (Japan).

Prof. Breckon is a Chartered Engineer, Chartered Scientist and a Fellow of the British Computer Society. In addition, he is an Accredited Senior Imaging Scientist and Fellow of the Royal Photographic Society. He led the development of image-based automatic threat detection for the 2008 UK MoD Grand Challenge winners [R.J. Mitchell Trophy, (2008), IET Innovation Award (2009)]. His work is recognised as a recipient of the Royal Photographic Society Selwyn Award for early-career contribution to imaging science (2011). http://www.durham.ac.uk/toby.breckon/