LEARNING FACTORIZED REPRESENTATIONS FOR OPEN-SET DOMAIN ADAPTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Domain adaptation for visual recognition has undergone great progress in the past few years. Nevertheless, most existing methods work in the so-called closed-set scenario, assuming that the classes depicted by the target images are exactly the same as those of the source domain. In this paper, we tackle the more challenging, yet more realistic case of open-set domain adaptation, where new, unknown classes can be present in the target data. While, in the unsupervised scenario, one cannot expect to be able to identify each specific new class, we aim to automatically detect which samples belong to these new classes and discard them from the recognition process. To this end, we rely on the intuition that the source and target samples depicting the known classes can be generated by a shared subspace, whereas the target samples from unknown classes come from a different, private subspace. We therefore introduce a framework that factorizes the data into shared and private parts, while encouraging the shared representation to be discriminative. Our experiments on standard benchmarks evidence that our approach significantly outperforms the state-of-the-art in open-set domain adaptation.

1 INTRODUCTION

In many practical machine learning scenarios, the test samples are drawn from a different distribution from the training ones, due to varying acquisition conditions, such as different data sources, illumination conditions and cameras, in the context of visual recognition. Over the years, great progress has been achieved to tackle this problem, known has the domain shift. In particular, many methods aim to align the source (i.e., training) and target (i.e., test) distributions by learning domain-invariant embeddings (Pan et al., 2011; Gong et al., 2012; Fernando et al., 2013; Sun et al., 2016), the most recent approaches relying on deep networks (Ganin & Lempitsky, 2014; Long et al., 2015; Bousmalis et al., 2016; Tzeng et al., 2017; Long et al., 2016a; Yan et al., 2017).

While effective, these methods work under the assumption that the source and target data contain exactly the same classes. In practice, however, this assumption may easily be violated, as the target data will often contain additional classes that were not present within the source data. For example, when training a model to recognize office objects from images, as with the popular Office dataset (Saenko et al., 2010), one should still expect to see new objects, unobserved during training, when deploying the model in the real world. While one should not expect the model to recognize the specific class of such objects, at least in unsupervised domain adaptation where no target labels are provided, it would nonetheless be beneficial to identify these objects as unknown instead of misclassifying them. This was the task addressed by Busto & Gall (2017) in their so-called *open-set* domain adaptation approach. This method aims to learn a mapping from the source samples to a subset of the target ones corresponding to those identified as coming from known classes. While reasonably effective, this procedure involves alternatively solving for the mapping and the assignment of the samples to known/unknown classes, which, as shown in our experiments, can be costly. Recently, Saito et al. (2018) introduced a deep learning framework for open-set domain adaptation, relying on adversarial training to separate the samples from the known classes from the unknown ones.

In this paper, we introduce a novel approach to open-set domain adaptation based on learning a factorized representation of the source and target data. In essence, we seek to model the samples from the known classes with a low-dimensional subspace, shared by the source and target domains, and the target samples from unknown classes with another subspace, specific to the target domain.

We then make use of group sparsity to encourage each target sample to be reconstructed by only one of these subspaces, which in turns lets us identify if this sample corresponds to a known or unknown class. We further show that we can obtain a more discriminative shared representation by jointly learning a linear classifier within our framework. Ultimately, our approach therefore allows us to jointly separate the target samples between known and unknown classes and represent the source and target samples within a consistent, shared latent space. Note that our approach is more intuitive than (Bousmalis et al., 2016) for the open-set DA scenario in the sense that we model each target sample as being generated by either the shared subspace or the private one, which is crucial to identify the target samples depicting unknown classes. By contrast, in (Bousmalis et al., 2016), each sample is encoded as a *mixture* of shared and private representations, which doesn't provide information to discriminate samples from unknown classes.

We demonstrate the effectiveness of our approach on several open set domain adaptation benchmarks for visual object recognition. Our method consistently and significantly outperforms the technique of Busto & Gall (2017) on all benchmarks, as well as the end-to-end learning approach of Saito et al. (2018) on the Office dataset, thus showing the benefits of learning shared and private representations corresponding to the known and unknown classes, respectively. Furthermore, it is faster than the algorithm of Busto & Gall (2017) by an order of magnitude.

2 Related Work

Domain adaptation for visual recognition has become increasingly popular over the past few years, in large part thanks to the benchmark Office dataset of Saenko et al. (2010). A natural approach to tackling the domain shift consists of learning a transformation of the data such that the distributions of the source and target samples are as similar as possible in the resulting space (Baktashmotlagh et al., 2014; 2013; Sun et al., 2016). Instead of learning a transformation of the data, other methods have been proposed to re-weight the source samples, so as to rely more strongly on those that look similar to the target ones (Quiñonero C. et al., 2009; Gong et al., 2013).

With the advent of deep learning for visual recognition, domain adaptation research also eventually turned to exploiting deep networks. While it was initially shown that deep features were more robust than handcrafted ones to the domain shift (Donahue et al., 2013), translating the above-mentioned distribution-matching ideas to end-to-end learning proved even more effective (Tzeng et al., 2014; Long et al., 2015; 2016b; Rozantsev et al., 2018; Sun & Saenko, 2016). In this context, other ideas were introduced, such as learning intermediate representations to interpolate between the source and target domains (Chopra S. & R., 2013; Tzeng et al., 2015), the use of adversarial domain classifiers (Ganin & Lempitsky, 2014; Tzeng et al., 2017), and additional reconstruction loss terms (Ghifary et al., 2016).

Despite achieving great progress to tackle the domain shift, all the aforementioned methods are designed for the closed-set scenario, where the source and target data depict the exact same set of classes. Inspired by recent advances in open-set recognition (Bendale & Boult, 2015; Scheirer et al., 2014), the work of Busto & Gall (2017) constitutes the first attempt to address the more realistic case where the target data contains samples from new, unknown classes. To achieve this, (Busto & Gall, 2017) proposed to jointly learn the assignments of the target samples to known/unknown classes and a mapping from the source data to the target samples depicting known classes. The resulting learning problem was solved by alternatively optimizing for the assignments and for the mapping, which can be costly. Very recently, a deep learning approach was proposed for open-set domain adaptation (Saito et al., 2018), relying on adversarial training to separate the unknown target samples from the known ones.

Here, we introduce a new solution to the open-set domain adaptation problem, where we model the source and target data with subspaces. Subspace-based representations have proven effective for domain adaptation (Gong et al., 2012; Gopalan et al., 2014; Fernando et al., 2013). Here, however, we exploit them in a different manner, based on the intuition that source samples and target samples from the known classes can be generated by a shared subspace, whereas target samples from unknown classes come from a private subspace. While the notion of shared-private representations has been exploited in the past, e.g., for multiview learning (Jia et al., 2010) and for closed-set domain adaptation (Bousmalis et al., 2016), the resulting techniques all use them to encode each sample as a mixture of shared and private information. By contrast, here, we aim to model each target sample

as being generated by either the shared subspace or the private one, which is crucial to identify the target samples depicting unknown classes.

Our experiments evidence that our open-set domain adaptation approach, based on shared-private representations, is more effective than the one of Busto & Gall (2017), consistently outperforming it on several datasets, and also faster by an order of magnitude. We also outperform the recent deep learning open-set domain adaptation framework of Saito et al. (2018) on the Office benchmark.

3 OUR APPROACH

The key idea behind our formulation is to find low-dimensional representations of the data, factorized into a subspace shared by the source samples and the target ones coming from known classes and another subspace specific to the target samples from unknown classes. Note that, when referring to target samples from known classes, we do not mean that these samples are labeled, but rather that they belong to the same set of classes as the source data. As a matter of fact, throughout the paper, we focus on the unsupervised domain adaptation scenario, where no target annotations are provided. In the remainder of this section, we first introduce the optimization problem at the heart of our approach, and then discuss two extensions of this basic formulation.

3.1 FRODA: FACTORIZED REPRESENTATIONS FOR OPEN-SET DOMAIN ADAPTATION

Given n_s source samples, grouped in a matrix $\boldsymbol{X}_s \in \mathbb{R}^{D \times n_s}$ and n_t target samples represented by $\boldsymbol{X}_t \in \mathbb{R}^{D \times n_t}$, our goal is to estimate a low-dimensional representation of each sample, such that the source and target samples coming from the same classes are generated by a shared subspace, whereas the target data from new, unknown classes are generated by a different, specific subspace. To this end, let $\boldsymbol{V} \in \mathbb{R}^{D \times d}$ be the matrix encoding the shared subspace, with $d \ll D$, and $\boldsymbol{U} \in \mathbb{R}^{D \times d}$ the one representing the private subspace. A naive approach to finding the low-dimensional representations of the data would involve solving

$$\min_{\boldsymbol{U},\boldsymbol{T},\boldsymbol{V},\boldsymbol{S}} \quad \|\boldsymbol{X}_t - \boldsymbol{B}\boldsymbol{T}\|_F^2 + \alpha\|\boldsymbol{X}_s - \boldsymbol{V}\boldsymbol{S}\|_F^2 \,, \tag{1}$$

where α sets the relative influence of both terms, $\boldsymbol{B} = [\boldsymbol{V}, \boldsymbol{U}] \in \mathbb{R}^{D \times 2d}$, and $\boldsymbol{T}$ and $\boldsymbol{S}$ encode the low-dimensional representations of the target and source data, respectively. This simple formulation, however, does not aim to separate the target samples belonging to known classes from the unknown ones, and thus will represent each target sample as a mixture of shared and private information.

Intuitively, we would rather like each target sample to be generated by either the shared subspace $\boldsymbol{V}$, or the private one $\boldsymbol{U}$. To address this, we propose to make use of a group sparsity regularizer on the coefficients of the target samples. Specifically, we split the coefficient vector $\boldsymbol{T}_i$ for target sample i into a part $\boldsymbol{T}_i^v$ that corresponds to the shared subspace and a part $\boldsymbol{T}_i^u$ that corresponds to the private one. We then encourage that either of these two parts goes to zero for each sample. To this end, we therefore write the optimization problem

$$\begin{aligned} \min_{\boldsymbol{U},\boldsymbol{T},\boldsymbol{V},\boldsymbol{S}} \quad & \|\boldsymbol{X}_t - \boldsymbol{B}\boldsymbol{T}\|_F^2 + \alpha\|\boldsymbol{X}_s - \boldsymbol{V}\boldsymbol{S}\|_F^2 + \lambda_1 \sum_{i=1}^{n_t} (\|\boldsymbol{T}_i^v\| + \|\boldsymbol{T}_i^u\|) \\ s.t. \quad & \sum_{j=1}^{d} \|\boldsymbol{U}_j\|^2 \leq 1, \sum_{j=1}^{d} \|\boldsymbol{V}_j\|^2 \leq 1 \,, \end{aligned} \tag{2}$$

where the constraints prevent the basis vectors of the subspaces from growing while the coefficients decrease (Lee et al., 2007), and where λ_1 is a scalar controlling the strength of the group sparsity regularizer. In essence, this formulation allows each target sample to be reconstructed from either the shared subspace or the target one, which reduces the influence of the samples from unknown classes on learning the shared representation.

Optimization. To solve equation 2 efficiently, we alternatively update one variable at a time while keeping the other ones fixed. Below, we describe the different updates.

Algorithm 1 : FRODA: Factorized Representations for Open-set Domain Adaptation

Input:
$\boldsymbol{X}_s \in \mathbb{R}^{D \times n_s}$: the source samples
$\boldsymbol{X}_t \in \mathbb{R}^{D \times n_t}$: the target samples
$d \ll D$: the dimensionality of the subspaces

Output:
$\boldsymbol{S} \in \mathbb{R}^{d \times n_s}, \boldsymbol{T} \in \mathbb{R}^{2d \times n_t}$

Initialize:
$\boldsymbol{V} \leftarrow PCA(\boldsymbol{X}_s)$
$\boldsymbol{U} \leftarrow Null(\boldsymbol{V})$ (i.e., truncated null space of $\boldsymbol{V}$)

1: Compute $\boldsymbol{T}$ from equation 5 by proximal gradient descent
2: Compute $\boldsymbol{S}$ by solving the linear least-squares problem $\min_{\boldsymbol{S}} \|\boldsymbol{X}_s - \boldsymbol{V}\boldsymbol{S}\|_F^2$
3: **repeat**
4: Compute $\boldsymbol{U}$ from equation 3 by the Lagrange dual method of (Lee et al., 2007)
5: Compute $\boldsymbol{V}$ from equation 4 by the Lagrange dual method of (Lee et al., 2007)
6: Compute $\boldsymbol{T}$ from equation 5 by proximal gradient descent
7: Compute $\boldsymbol{S}$ by solving the linear least-squares problem $\min_{\boldsymbol{S}} \|\boldsymbol{X}_s - \boldsymbol{V}\boldsymbol{S}\|_F^2$
8: **until** convergence

$\boldsymbol{B}$-minimization: Given the coefficients $\boldsymbol{S}$ and $\boldsymbol{T}$, we update the tuple $(\boldsymbol{U}, \boldsymbol{V})$ by solving

$$\min_{\boldsymbol{U}} \quad \|\boldsymbol{A} - \boldsymbol{U}\boldsymbol{T}^u\|_F^2 \quad s.t. \sum_{j=1}^{d} \|\boldsymbol{U}_j\|^2 \leq 1 \ , \tag{3}$$

with $\boldsymbol{A} = \boldsymbol{X}_t - \boldsymbol{V}\boldsymbol{T}^v$, and

$$\min_{\boldsymbol{V}} \quad \|\boldsymbol{A'} - \boldsymbol{V}\boldsymbol{T}^v\|_F^2 + \alpha\|\boldsymbol{X}_s - \boldsymbol{V}\boldsymbol{S}\|_F^2 \quad s.t. \sum_{j=1}^{d} \|\boldsymbol{V}_j\|^2 \leq 1 \ , \tag{4}$$

with $\boldsymbol{A'} = \boldsymbol{X}_t - \boldsymbol{U}\boldsymbol{T}^u$. These two sub-problems can be solved efficiently using the Lagrange dual formulation introduced in (Lee et al., 2007) to update the basis in a standard sparse coding context.

$\boldsymbol{T}$-minimization: Minimizing equation 2 with respect to $\boldsymbol{T}$, with all other parameters fixed, yields

$$\min_{\boldsymbol{T}} \quad \|\boldsymbol{X}_t - \boldsymbol{B}\boldsymbol{T}\|_F^2 + \lambda_1 \sum_{i=1}^{n_t} (\|\boldsymbol{T}_i^v\| + \|\boldsymbol{T}_i^u\|) \ , \tag{5}$$

which can be solved efficiently via the proximal gradient method (Mairal et al., 2014). In other words, for each target sample, we obtain $\boldsymbol{T}$ using group sparse coding to determine to which representation, shared or private, each target sample belongs.

$\boldsymbol{S}$-minimization: Solving equation 2 with respect to $\boldsymbol{S}$, with all the other parameters fixed, reduces to a linear least-squares problem, which has a closed-form solution.

To start optimization, we initialize $\boldsymbol{V}$ as the PCA subspace of the source data, and take $\boldsymbol{U}$ as its truncated null space. We then obtain the corresponding $\boldsymbol{T}$ and $\boldsymbol{S}$ as described above and start iterating. The pseudo-code of FRODA is provided in Algorithm 1. The steps are repeated until convergence, which typically occurs around 50 iterations, with each iteration taking roughly 0.05s.

Inference. After obtaining the final $2d$-dimensional representation of $\boldsymbol{T}$ for target samples, we determine if each sample i belongs to known classes or unknown ones based on the coefficients $\boldsymbol{T}_i^u$ and $\boldsymbol{T}_i^v$. More specifically, given a threshold ε, a target sample is assigned to the unknown classes if $\|\boldsymbol{T}_i^v\|/\|\boldsymbol{T}_i^u\| \leq \varepsilon$, which suggests that it can be well-reconstructed by the private subspace. In the presence of C known classes, we then train a $(C+1)$-way classifier by augmenting the d-dimensional source data $\boldsymbol{S}$, i.e. $\boldsymbol{S} \in \mathbb{R}^{d \times n_s}$, with the target samples $\boldsymbol{T}^u$ identified as unknown.

3.2 D-FRODA: DISCRIMINATIVE FRODA

The formulation above does not make use of the source labels at all during the representation learning stage. As such, it does not encourage the representation to be discriminative. To overcome this, we extend our basic formulation to further account for the classification task at hand. Specifically, let $\boldsymbol{L} = [\boldsymbol{l}_1 \dots \boldsymbol{l}_{n_s}] \in \mathbb{R}^{C \times n_s}$ be the matrix containing the source labels, where $\boldsymbol{l}_i \in \mathbb{R}^C$ represents the one-hot encoding of the label of sample i. We then write our D-FRODA formulation as

$$\begin{aligned} \min_{\boldsymbol{U},\boldsymbol{T},\boldsymbol{V},\boldsymbol{S},\boldsymbol{W}} \quad & \|\boldsymbol{X}_t - \boldsymbol{B}\boldsymbol{T}\|_F^2 + \alpha\|\boldsymbol{X}_s - \boldsymbol{V}\boldsymbol{S}\|_F^2 + \beta\|\boldsymbol{L} - \boldsymbol{W}\boldsymbol{S}\|_F^2 + \lambda_1 \sum_{i=1}^{n_t} (\|T_i^v\| + \|T_i^u\|)) \\ s.t. \quad & \sum_{j=1}^{d} \|\boldsymbol{U}_j\|^2 \leq 1, \sum_{j=1}^{d} \|\boldsymbol{V}_j\|^2 \leq 1 \ , \end{aligned} \tag{6}$$

where $\boldsymbol{W} \in \mathbb{R}^{C \times d}$ is the matrix containing the parameters of a linear classifier for the source data.

Optimization. To optimize equation 6, we follow a similar alternating strategy as before. The $\boldsymbol{B}$-minimization and $\boldsymbol{T}$-minimization steps are unchanged, but the $\boldsymbol{S}$-minimization now incorporates a new term and we further need to solve for the classifier parameters $\boldsymbol{W}$. This translates to:

$\boldsymbol{S}$-minimization: Minimizing equation 6 with respect to $\boldsymbol{S}$, with all the other parameters fixed, still reduces to a linear least-squares problem. The two terms involving $\boldsymbol{S}$ can be grouped into a single one of the form $\|\boldsymbol{X}_{new} - \boldsymbol{V}_{new}\boldsymbol{S}\|_F^2$, where $\boldsymbol{X}_{new} = \begin{pmatrix} \sqrt{\alpha}\boldsymbol{X}_s \\ \sqrt{\beta}\boldsymbol{L} \end{pmatrix}$ and $\boldsymbol{V}_{new} = \begin{pmatrix} \sqrt{\alpha}\boldsymbol{V} \\ \sqrt{\beta}\boldsymbol{W} \end{pmatrix}$, and thus $\boldsymbol{S}$ can be obtained in closed form.

$\boldsymbol{W}$-minimization: With all the other parameters fixed, finding $\boldsymbol{W}$ corresponds to a linear least-squares problem, with a closed-form solution.

Inference. The same inference strategy as before can be followed to label the target samples. Another option here is to make use of $\boldsymbol{W}$ to classify the samples identified as belonging to known classes. We compare these two strategies in our experiments.

3.3 D-FRODA-U: D-FRODA WITH UNKNOWN SOURCE CLASSES

Until now, we have tackled the scenario where there are no unknown classes in the source data, which we believe corresponds to the typical application scenario, since the source data can in general be fully annotated. Nevertheless, to match the scenario of Busto & Gall (2017), that assumes to have access to additional source samples from unknown classes, yet different from the target unknown classes, we introduce a modified version of our approach that takes such auxiliary data into account. Note that, since one knows which source samples are from unknown classes, it is also possible to simple discard them from training. To nonetheless handle them, we re-write equation 6 as

$$\begin{aligned} \min_{\boldsymbol{U},\boldsymbol{T},\boldsymbol{V},\boldsymbol{U'},\boldsymbol{S'},\boldsymbol{W'}} \quad & \|\boldsymbol{X}_t - \boldsymbol{B}\boldsymbol{T}\|_F^2 + \alpha\|\boldsymbol{X'}_s - \boldsymbol{B'}\boldsymbol{S'}\|_F^2 + \beta\|\boldsymbol{L} - \boldsymbol{W'}\boldsymbol{S'}\|_F^2 \\ & + \lambda_1 \sum_{i=1}^{n_t} (\|\boldsymbol{T}_i^v\| + \|\boldsymbol{T}_i^u\|) + \lambda_2 \sum_{i=1}^{n_s} \left(\|\boldsymbol{S'}_i^v\| + \|\boldsymbol{S'}_i^u\|\right) \\ s.t. \quad & \sum_{j=1}^{2d} \|\boldsymbol{B}_j\|^2 \leq 1, \sum_{j=1}^{2d} \|\boldsymbol{B'}_j\|^2 \leq 1 \ , \end{aligned} \tag{7}$$

where $\boldsymbol{X'}_s$ contains the source samples from both known and unknown classes, and $\boldsymbol{B'} = [\boldsymbol{V}, \boldsymbol{U'}] \in \mathbb{R}^{D \times 2d}$ denotes the source transformation matrix with $\boldsymbol{U'} \in \mathbb{R}^{D \times d}$ the private subspace for the source data. Note that, similarly to the target coefficients, we have now separated the source coefficients for each sample $\boldsymbol{S'}_i$ into a part corresponding to the shared subspace $\boldsymbol{S'}_i^v$ and a part corresponding to the private one $\boldsymbol{S'}_i^u$. Note also that the classifier parameters $\boldsymbol{W'}$ now account for $C+1$ classes, the additional class corresponding to the unknown samples.

Optimization. We follow a similar iterative procedure to the one used before, with modifications to update $\boldsymbol{B'}$ and $\boldsymbol{S'}$, as discussed below.

Table 1: Recognition accuracies on the 12 source/target pairs of the BCIS dataset (Tommasi & Tuytelaars, 2014) using a linear SVM classifier. **B**: Bing, **C**: Caltech256, **I**: ImageNet, **S**: SUN.

Method	$\mathbf{B} \rightarrow \mathbf{C}$	$\mathbf{B} \rightarrow \mathbf{I}$	$\mathbf{B} \rightarrow \mathbf{S}$	$\mathbf{C} \rightarrow \mathbf{B}$	$\mathbf{C} \rightarrow \mathbf{I}$	$\mathbf{C} \rightarrow \mathbf{S}$
TCA (Pan et al., 2011)	62.8 ± 3.8	56.6 ± 4.5	29.6 ± 4.2	38.9 ± 1.9	60.2 ± 1.4	29.7 ± 1.6
GFK (Gong et al., 2012)	66.2 ± 4.0	58.3 ± 3.1	23.8 ± 2.0	40.2 ± 1.8	62.2 ± 1.5	28.5 ± 1.0
SA (Fernando et al., 2013)	66.0 ± 3.4	57.8 ± 3.2	24.3 ± 2.6	40.3 ± 1.7	62.5 ± 0.8	29.0 ± 1.5
CORAL (Sun et al., 2016)	68.8 ± 3.3	60.9 ± 2.6	27.2 ± 3.9	40.7 ± 1.5	64.0 ± 2.6	31.4 ± 0.8
ATI (Busto & Gall, 2017)	71.4 ± 2.3	69.0 ± 2.8	37.4 ± 2.6	45.7 ± 3.0	67.9 ± 4.2	37.5 ± 2.7
AODA (Saito et al., 2018)	$\mathbf{76.2 \pm 1.7}$	70.9 ± 3.2	$\mathbf{57.3 \pm 1.1}$	63.5 ± 2.1	73.5 ± 0.8	60.5 ± 0.8
FRODA	73.8 ± 6.1	71.0 ± 2.0	54.7 ± 2.9	67.5 ± 1.4	74.5 ± 1.7	61.6 ± 2.2
D-FRODA	74.6 ± 5.5	$\mathbf{71.4 \pm 2.0}$	55.4 ± 2.7	$\mathbf{67.6 \pm 1.2}$	$\mathbf{75.0 \pm 1.8}$	$\mathbf{61.7 \pm 2.1}$

Method	$\mathbf{I} \rightarrow \mathbf{B}$	$\mathbf{I} \rightarrow \mathbf{C}$	$\mathbf{I} \rightarrow \mathbf{S}$	$\mathbf{S} \rightarrow \mathbf{B}$	$\mathbf{S} \rightarrow \mathbf{C}$	$\mathbf{S} \rightarrow \mathbf{I}$	Avg.
TCA (Pan et al., 2011)	40.9 ± 2.9	68.6 ± 1.8	34.5 ± 3.8	19.4 ± 2.1	32.0 ± 3.9	31.1 ± 4.6	42
GFK (Gong et al., 2012)	42.6 ± 2.4	73.3 ± 3.6	32.7 ± 3.6	16.9 ± 1.5	28.6 ± 3.8	26.4 ± 1.1	41.6
SA (Fernando et al., 2013)	43.1 ± 1.6	72.8 ± 3.1	32.2 ± 3.7	17.5 ± 1.6	29.2 ± 4.2	27.1 ± 1.3	41.8
CORAL (Sun et al., 2016)	44.6 ± 2.5	74.5 ± 3.4	35.4 ± 4.4	18.7 ± 1.2	33.6 ± 5.3	31.3 ± 1.3	44.3
ATI (Busto & Gall, 2017)	48.8 ± 2.3	77.5 ± 2.2	43.4 ± 4.8	23.2 ± 3.2	47.3 ± 2.9	33.0 ± 1.1	50.2
AODA (Saito et al., 2018)	66.3 ± 0.9	78.1 ± 0.9	59.4 ± 1.4	$\mathbf{56.5 \pm 2.6}$	59.6 ± 3.1	$\mathbf{63.2 \pm 1.3}$	65.4
FRODA	66.0 ± 1.9	79.9 ± 1.7	59.2 ± 2.1	55.7 ± 2.5	$\mathbf{61.2 \pm 1.8}$	59.4 ± 1.9	65.4
D-FRODA	$\mathbf{66.4 \pm 1.7}$	$\mathbf{80.5 \pm 1.6}$	$\mathbf{59.8 \pm 2.0}$	55.5 ± 2.4	$\mathbf{61.2 \pm 1.9}$	59.6 ± 2.2	**65.7**

$\boldsymbol{S'}$-minimization: To minimize equation 7 w.r.t. $\boldsymbol{S'}$, with all other parameters fixed, we write

$$\min_{\boldsymbol{S'}} \quad \alpha \|\boldsymbol{X'}_s - \boldsymbol{B'S'}\|_F^2 + \beta \|\boldsymbol{L} - \boldsymbol{W'S'}\|_F^2 + \lambda_2 \sum_{i=1}^{n_s} \left(\|\boldsymbol{S'}_i^v\| + \|\boldsymbol{S'}_i^u\| \right) \ . \tag{8}$$

The first two terms can be grouped into a single squared Frobenius norm, thus resulting in a sparse group lasso problem, which, as when updating $\boldsymbol{T}$ in FRODA, can be solved via proximal gradient descent (Mairal et al., 2014).

$\boldsymbol{B'}$-minimization: Given the coefficients $\boldsymbol{S'}^v$ and $\boldsymbol{S'}^u$, we can update $\boldsymbol{U'}$ by solving

$$\min_{\boldsymbol{U'}} \quad \|\boldsymbol{A} - \boldsymbol{U'S'}^u\|_F^2 \quad s.t. \sum_{j=1}^{d} \|\boldsymbol{U'}_j\|^2 \leq 1 \ , \tag{9}$$

with $\boldsymbol{A} = \boldsymbol{X'}_s - \boldsymbol{V}\boldsymbol{S'}^v$, and $\boldsymbol{V}$ by solving

$$\min_{\boldsymbol{V}} \quad \|\boldsymbol{A'} - \boldsymbol{VC}\|_F^2 \quad s.t. \sum_{j=1}^{d} \|\boldsymbol{V}_j\|^2 \leq 1 \ , \tag{10}$$

with $\boldsymbol{A'} = \begin{pmatrix} \boldsymbol{X}_t - \boldsymbol{UT}^u \\ \sqrt{\alpha}(\boldsymbol{X'}_s - \boldsymbol{U'S'}^u) \end{pmatrix}$ and $\boldsymbol{C} = \begin{pmatrix} \boldsymbol{T}^v \\ \sqrt{\alpha}\boldsymbol{S'}^v \end{pmatrix}$. As in FRODA, these two sub-problems can be solved efficiently using the Lagrange dual formulation of Lee et al. (2007).

4 Experiments

We evaluate our approach on the task of open-set visual domain adaptation using two benchmark datasets, and compare its performance against the state-of-the-art open-set domain adaptation methods on each dataset.[1] Note that we also report the results of the methods used as baselines in (Busto & Gall, 2017). For a dataset with C source classes, we report the accuracy on $C + 1$ classes, the additional one corresponding to the unknown case.

Implementation details. Following Busto & Gall (2017), we represent the source and target samples with 4096-dimensional $DeCAF_7$ features (Donahue et al., 2013) and first reduce their dimensionality by performing PCA jointly on the source and target data and keeping the components encoding 99% of the data variance. To then determine the dimensionality d of our shared and private subspaces, we make use of the subspace disagreement measure of (Gong et al., 2012). For all our experiments, the hyperparameters of our approach were set as follows: $\alpha = 0.1$, $\beta = 0.01$, $\lambda = 0.001$ and $\varepsilon = 0.2$. For recognition, for the comparison with (Busto & Gall, 2017) to be fair, we employ a linear SVM classifier in a one-vs-one fashion. Nevertheless, we also report results obtained with a $k-$Nearest-Neighbor classifier (with $k = 3$) and with our linear classifier with parameters $\boldsymbol{W}$ learnt during training.

[1] Among the different variants of ATI (Busto & Gall, 2017), we report the best one in each experiment.

Table 2: Recognition accuracies of variants of our approach using linear SVM, nearest neighbor (NN) and our linear classifier ($\boldsymbol{W}$) on the 12 source/target pairs of the BCIS dataset (Tommasi & Tuytelaars, 2014). **B**: Bing, **C**: Caltech256, **I**: ImageNet, **S**: SUN.

Method	**B → C**	**B → I**	**B → S**	**C → B**	**C → I**	**C → S**
FRODA-SVM	73.8 ± 6.1	71.0 ± 2.0	54.7 ± 2.9	67.5 ± 1.4	74.5 ± 1.7	61.6 ± 2.2
FRODA-NN	67.7 ± 2.8	64.1 ± 2.4	54.4 ± 3.5	65.1 ± 2.9	72.9 ± 1.7	60.5 ± 2.0
D-FRODA-SVM	$\mathbf{74.6 \pm 5.5}$	$\mathbf{71.4 \pm 2.0}$	55.4 ± 2.7	$\mathbf{67.6 \pm 1.2}$	$\mathbf{75.0 \pm 1.8}$	$\mathbf{61.7 \pm 2.1}$
D-FRODA-W	61.2 ± 1.2	59.3 ± 1.1	53.3 ± 3.0	63.1 ± 0.9	66.2 ± 1.7	60.2 ± 2.1
D-FRODA-NN	67.7 ± 3.3	63.3 ± 2.8	54.0 ± 3.5	65.4 ± 2.8	73.4 ± 1.5	60.3 ± 2.4
D-FRODA-U-SVM	71.9 ± 3.8	69.2 ± 2.7	$\mathbf{56.7 \pm 3.3}$	64.8 ± 1.8	72.8 ± 2.7	59.4 ± 2.3
D-FRODA-U-W	55.9 ± 3.6	55.5 ± 4.5	41.7 ± 4.8	52.6 ± 4.7	61.4 ± 3.6	49.5 ± 4.1
D-FRODA-U-NN	57.6 ± 9.1	52.2 ± 5.0	47.5 ± 6.5	51.8 ± 5.7	64.0 ± 5.7	57.7 ± 5.6

Method	**I → B**	**I → C**	**I → S**	**S → B**	**S → C**	**S → I**	Avg.
FRODA-SVM	66.0 ± 1.9	79.9 ± 1.7	59.2 ± 2.1	55.7 ± 2.5	61.2 ± 1.8	59.4 ± 1.9	65.4
FRODA-NN	60.9 ± 3.7	77.7 ± 2.8	58.0 ± 2.2	53.4 ± 2.2	61.2 ± 1.4	58.1 ± 1.5	62.8
D-FRODA-SVM	$\mathbf{66.4 \pm 1.7}$	$\mathbf{80.5 \pm 1.6}$	$\mathbf{59.8 \pm 2.0}$	55.5 ± 2.4	61.2 ± 1.9	59.6 ± 2.2	**65.7**
D-FRODA-W	62.2 ± 1.6	70.5 ± 2.5	58.1 ± 1.7	$\mathbf{56.4 \pm 1.9}$	58.9 ± 1.5	58.5 ± 0.7	60.7
D-FRODA-NN	60.9 ± 4.1	78.7 ± 2.8	57.7 ± 2.0	53.0 ± 2.3	61.2 ± 1.2	57.9 ± 1.6	62.8
D-FRODA-U-SVM	66.0 ± 1.2	76.8 ± 1.9	57.2 ± 4.5	56.3 ± 1.9	$\mathbf{61.8 \pm 3.0}$	$\mathbf{59.9 \pm 1.7}$	64.4
D-FRODA-U-W	54.6 ± 4.4	66.2 ± 3.8	43.9 ± 5.4	47.6 ± 3.8	53.5 ± 4.9	51.3 ± 3.7	52.8
D-FRODA-U-NN	58.4 ± 6.1	70.9 ± 4.5	52.8 ± 9.0	55.4 ± 2.0	61.5 ± 2.0	60.1 ± 1.7	57.5

Results on the dense cross-dataset benchmark. We first evaluate our approach on the challenging cross-dataset benchmark of Tommasi & Tuytelaars (2014). This dataset was built using images depicting 40 object categories and coming from four datasets, namely Bing (B), Caltech256 (C), ImageNet (I) and SUN (S), hence referred to as BCIS. Following Busto & Gall (2017), we consider the samples from the first 10 classes as known instances, while the samples with class labels $11, 12, \cdots, 25$ and $26, 27, \cdots, 40$ are taken to be the unknown samples in the source and target domains, respectively. We follow the unsupervised protocol of Tommasi & Tuytelaars (2014), which relies on 50 source samples per class and 30 target images per class, except when the target data is coming from SUN, in which case only 20 images per class are employed. Note that only the $DeCAF_7$ features are publicly available. To nonetheless evaluate the AODA method of Saito et al. (2018), we made use of a network with two fully-connected layers, with 1024 and 128 units, respectively, and a final classification layer, which takes the $DeCAF_7$ features as input.

In Table 1, we compare the results of our methods with those of the baselines on all 12 domain pairs of this dataset. Note that our algorithms (both with and without the discriminative term) significantly outperform all the baselines, and in particular the state-of-the-art one of Busto & Gall (2017) by a large margin. For instance, the margin exceeds 32, resp. 26, percentage points when going from SUN to Bing and ImageNet, respectively. This, we believe, clearly evidences the benefits of our factorized representations, which allow us to separate the unknown target samples from the ones coming from known classes, thus yielding a better representation for the known classes.

In Table 2, we compare different versions of our method, corresponding to using different classifiers and to using additional unknown source data. Note that the linear SVM classifier, when used with our framework, tends to perform the best, followed by the NN one and finally the learnt linear classifier. This, we believe, can be explained by the fact that, while the linear classifier helps learning a more discriminative representation, it remains less powerful than the other two classifiers to label the target samples. Note also that the use of unknown source data does not consistently help in our framework. Nevertheless, the corresponding results still outperform those of Busto & Gall (2017).

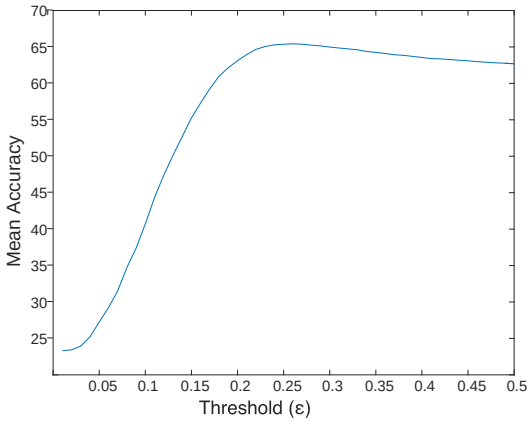

Figure 1: Sensitivity to ε

We further evaluate the robustness of our approach to the choice of threshold ε to separate the target samples from known/unknown classes.

Table 3: Recognition accuracies on the 6 source/target pairs of the Office dataset (Saenko et al., 2010) using a linear SVM classifier. **A**: Amazon, **W**: Webcam, **D**: DSLR.

Method	**A** → **D**	**A** → **W**	**W** → **A**	**W** → **D**	**D** → **A**	**D** → **W**	Avg.
LSVM	72.6	57.5	49.2	98.8	45.1	88.5	68.6
DAN (Long et al., 2016a)	77.6	72.5	60.8	98.3	57	88.4	75.8
RTN (Long et al., 2016b)	76.6	73	62.4	98.8	57.2	89	76.2
BP (Ganin & Lempitsky, 2014)	78.3	75.9	64	98.7	57.6	89.8	77.4
ADDA (Tzeng et al., 2017)	52.5	58.3	54.1	89.1	45.3	79.1	63.1
DSN (Bousmalis et al., 2016)	58.3	57.2	55.1	79.3	58.1	70.2	63.0
ATI (Busto & Gall, 2017)	79.8	78.4	76.7	**98.8**	71.3	94.4	83.2
AODA (Saito et al., 2018)	76.6	74.9	**81.2**	96.9	62.3	**94.6**	81.1
FRODA	**88.0**	**78.7**	76.5	98.0	**73.7**	**94.6**	84.9
D-FRODA	87.4	78.1	77.1	98.5	73.6	94.4	**84.9**

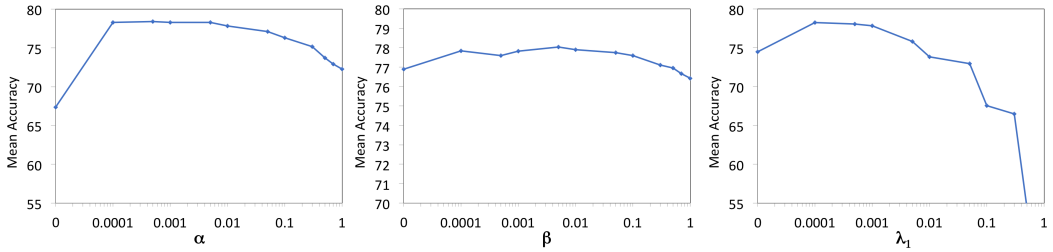

Figure 2: Sensitivity to α, β, and λ_1.

In the figure on the right, we plot the average accuracy over all 12 pairs of the BCIS dataset as a function of the value of ε. Note that, once a sufficiently large threshold is reached, the results are quite stable. This indicates that our algorithm is robust to the specific value of this hyperparameter.

Results on the Office dataset. We further evaluate our approach on the slightly less challenging, although standard Office benchmark (Saenko et al., 2010). This dataset contains three different domains, namely Amazon (A), DSLR (D) and Webcam (W), sharing 31 object categories, but differing in data acquisition process. As in (Busto & Gall, 2017), we take all the samples from the first 10 classes to represent the known ones, and all the samples with class labels $11, 12, \cdots, 20$ and $21, 22, \cdots, 31$ as unknown source and target data, respectively.

We report the results of our algorithms and of the baselines for all 6 domain pairs of this dataset in Table 3. As before, note that we outperform the baselines in this open-set scenario. This includes the end-to-end AlexNet-based approach of Saito et al. (2018) for open-set domain adaptation, as well as the state-of-the-art UDA methods of Tzeng et al. (2017) and Bousmalis et al. (2016), the latter of which is closest in spirit to our approach. In Table 4, we compare the different variants of our approach. The conclusions that one can draw from these results are similar to those for the BCIS dataset, thus showing that our method generalizes well across different domain adaptation datasets.

To evaluate the robustness of our method to the hyper-parameters of α, β, and λ, in Fig. 2, we plot the average accuracy over all 6 pairs of the Office dataset as a function of the value of α, β, and λ. Note that our results are stable for large ranges of these values.

Runtimes. As mentioned in Section 3, one iteration of our approach takes on average 0.05 second, and our algorithm typically takes around 50 iterations to converge. This yields a total runtime of roughly 2.5 seconds. By contrast, the publicly available implementation of the method of Busto & Gall (2017) takes on average 8 seconds per iteration and typically converges in 4 iterations, leading to a total runtime of roughly 32 seconds. Note that these runtimes were measured on the same computer and that both methods rely on the same input features. Therefore, our approach is not only significantly more accurate than (Busto & Gall, 2017), but also faster by an order of magnitude.

5 Conclusion

We have introduced a novel approach to open-set domain adaptation, based on the intuition that source and target samples coming from the same, known classes can be represented by a shared sub-

Table 4: Recognition accuracies of variants of our approaches using a linear SVM, nearest neighbor (NN) and our linear classifier ($\boldsymbol{W}$) on the 6 source/target pairs of the Office dataset (Saenko et al., 2010).

Method	$\mathbf{A} \rightarrow \mathbf{D}$	$\mathbf{A} \rightarrow \mathbf{W}$	$\mathbf{W} \rightarrow \mathbf{A}$	$\mathbf{W} \rightarrow \mathbf{D}$	$\mathbf{D} \rightarrow \mathbf{A}$	$\mathbf{D} \rightarrow \mathbf{W}$	Avg.
FRODA-SVM	**88.0**	78.7	76.5	98.0	**73.7**	**94.6**	**84.9**
FRODA-NN	83.9	69.5	75.0	97.7	69.0	83.9	79.8
D-FRODA-SVM	87.4	78.1	**77.1**	**98.5**	73.6	94.4	84.9
D-FRODA-NN	83.9	70.1	75.1	96.8	69.2	84.5	79.9
D-FRODA-W	71.1	65.3	68.1	83.0	67.3	79.1	72.3
D-FRODA-U-SVM	81.9	**83.5**	75.5	96.2	70.6	94.2	83.7
D-FRODA-U-NN	78.1	72.1	69.1	93.6	67.0	75.7	75.9
D-FRODA-U-W	73.4	65.9	62.8	88.0	61.7	65.1	69.5

space, while target samples from unknown classes should be modeled with a private subspace. Each step of the resulting algorithms can be solved efficiently. As demonstrated by our experiments, our method outperforms the state of the art in open-set domain adaptation and is one order of magnitude faster than the technique of Busto & Gall (2017). We believe that this clearly evidences the benefits of learning factorized representations, which allows us to jointly discard the unknown target samples and learn a better shared representation. In the future, we will investigate ways to make better use of unknown source data, and to exploit more effective classifiers, such as SVM, directly within our D-FRODA formulation.

REFERENCES

M. Baktashmotlagh, M. Harandi, B. Lovell, and M. Salzmann. Unsupervised domain adaptation by domain invariant projection. In *Proc. Int. Conference on Computer Vision*, 2013.

M. Baktashmotlagh, M. Harandi, B. Lovell, and M. Salzmann. Domain adaptation on statistical manifold. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

A. Bendale and T. Boult. Towards open world recognition. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

K. Bousmalis, G. Trigeorgis, N. Silberman, D. Krishnan, and D. Erhan. Domain separation networks. In *Proc. Advances in Neural Information Processing Systems*, 2016.

P. Busto and J. Gall. Open set domain adaptation. In *Proc. Int. Conference on Computer Vision*, 2017.

Balakrishnan S. Chopra S. and Gopalan R. Dlid: Deep learning for domain adaptation by interpolating between domains. In *ICML workshop on Challenges in Representation Learning*, 2013.

J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *Proc. Int. Conference on Machine Learning*, 2013.

B. Fernando, A. Habrard, M. Sebban, and T. Tuytelaars. Unsupervised visual domain adaptation using subspace alignment. In *Proc. Int. Conference on Computer Vision*, 2013.

Y. Ganin and V. Lempitsky. Unsupervised domain adaptation by backpropagation. *arXiv preprint arXiv:1409.7495*, 2014.

M. Ghifary, Bastiaan W. Kleijn, M. Zhang, D. Balduzzi, and W. Li. Deep reconstruction-classification networks for unsupervised domain adaptation. In *Proc. European Conference on Computer Vision*, 2016.

B. Gong, Y. Shi, F. Sha, and K. Grauman. Geodesic flow kernel for unsupervised domain adaptation. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

B. Gong, K. Grauman, and F. Sha. Connecting the dots with landmarks: Discriminatively learning domain-invariant features for unsupervised domain adaptation. In *Proc. Int. Conference on Machine Learning*, 2013.

R. Gopalan, R. Li, and R. Chellappa. Unsupervised adaptation across domain shifts by generating intermediate data representations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2014.

Y. Jia, M. Salzmann, and T. Darrell. Factorized latent spaces with structured sparsity. In *Proc. Advances in Neural Information Processing Systems*, 2010.

H. Lee, A. Battle, R. Raina, and A. Ng. Efficient sparse coding algorithms. In *Proc. Advances in Neural Information Processing Systems*, 2007.

M. Long, Y. Cao, J. Wang, and M. Jordan. Learning transferable features with deep adaptation networks. *arXiv preprint arXiv:1502.02791*, 2015.

M. Long, J. Wang, and M. Jordan. Deep transfer learning with joint adaptation networks. corr, vol. *arXiv preprint arXiv:1605.06636*, 2016a.

M. Long, H. Zhu, J. Wang, and M. Jordan. Unsupervised domain adaptation with residual transfer networks. In *Proc. Advances in Neural Information Processing Systems*, 2016b.

J. Mairal, F. Bach, J. Ponce, G. Sapiro, R. Jenatton, and G. Obozinski. Spams: A sparse modeling software, v2. 3. *URL http://spams-devel. gforge. inria. fr/downloads.html*, 2014.

S. Pan, I. Tsang, J. Kwok, and Q. Yang. Domain adaptation via transfer component analysis. *IEEE Transactions on Neural Networks*, 2011.

J Quiñonero C., M. Sugiyama, A. Schwaighofer, and N. Lawrence. Covariate shift by kernel mean matching. *Dataset Shift in Machine Learning*, 2009.

A. Rozantsev, M. Salzmann, and P. Fua. Beyond sharing weights for deep domain adaptation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018.

K. Saenko, B. Kulis, M. Fritz, and T. Darrell. Adapting visual category models to new domains. In *Proc. European Conference on Computer Vision*, 2010.

K. Saito, S. Yamamoto, Y. Ushiku, and T. Harada. Open set domain adaptation by backpropagation. *Proc. European Conference on Computer Vision*, 2018.

W. J Scheirer, L. Jain, and T. Boult. Probability models for open set recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2014.

B. Sun and K. Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *Proc. European Conference on Computer Vision*, 2016.

B. Sun, J. Feng, and K. Saenko. Return of frustratingly easy domain adaptation. In *AAAI Conference on Artificial Intelligence*, 2016.

T. Tommasi and T. Tuytelaars. A testbed for cross-dataset analysis. In *Proc. European Conference on Computer Vision*, 2014.

E. Tzeng, J. Hoffman, N. Zhang, K. Saenko, and T. Darrell. Deep domain confusion: Maximizing for domain invariance. *arXiv preprint arXiv:1412.3474*, 2014.

E. Tzeng, J. Hoffman, T. Darrell, and K. Saenko. Simultaneous deep transfer across domains and tasks. In *Proc. Int. Conference on Computer Vision*, 2015.

E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell. Adversarial discriminative domain adaptation. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

H. Yan, Y. Ding, P. Li, Q. Wang, Y. Xu, and W. Zuo. Mind the class weight bias: Weighted maximum mean discrepancy for unsupervised domain adaptation. *arXiv preprint arXiv:1705.00609*, 2017.