Analyzing the Effects of Dimensionality Reduction for Unsupervised Domain Adaptation

Renato Sergio Lopes Junior
Smart Sense Laboratory
Department of Computer Science
Federal University of Minas Gerais (UFMG)
Belo Horizonte, Minas Gerais, Brazil
Email: renato.junior@dcc.ufmg.br

William Robson Schwartz
Smart Sense Laboratory
Department of Computer Science
Federal University of Minas Gerais (UFMG)
Belo Horizonte, Minas Gerais, Brazil
Email: william@dcc.ufmg.br

***Abstract*—Deep neural networks are extensively used for solving a variety of computer vision problems. However, in order for these networks to obtain good results, a large amount of data is necessary for training. In image classification, this training data consists of images and labels that indicate the class portrayed by each image. Obtaining this large labeled dataset is very time and resource consuming. Therefore, domain adaptation methods allow different, but semantic-related, datasets that are already labeled to be used during training, thus eliminating the labeling cost. In this work, the effects of embedding dimensionality reduction in a state-of-the-art domain adaptation method are analyzed. Furthermore, we experiment with a different approach that use the available data from all domains to compute the confidence of pseudo-labeled samples. We show through experiments in commonly used datasets that, in fact, the proposed modifications led to better results in the target domain in some scenarios.**

I. Introduction

In recent times, the use of smart systems to automate different tasks in various areas has been on the rise. Computer vision and machine learning techniques are extensively used in these systems to deal with tasks concerning visual data [1]. One such task is image classification, in which images are classified based on the class portrayed by them, for example classifying which product is in a picture taken by a user on their smartphone camera. In order to obtain an accurate classification model, we need a large amount of labeled data that are used during training. Labeling data is often a very time and resource consuming step, as it usually must be done manually by a human. As large repositories of publicly-accessible data are available online due to the *big-data phenomenon* of the recent years [2], domain adaptation methods will allow labeled data that are available from a different, but semantically related, dataset to be used during training, thus eliminating the high cost of labeling data and making the development of smart computer systems more time and cost efficient.

Deep convolutional neural networks, like the ResNets [3] and DenseNets [4], [5], are extensively employed in many computer vision tasks, such as image classification, semantic segmentation and object detection, due to the remarkable results achieved by them [2]. In image classification, a large labeled dataset $\{(x_i, y_i)\}_{i=1}^{N}$, where x_i is an image and $y_i \in \{1, 2, ..., K\}$ is a label that indicates the class portrayed by the i-th image, is used to train these networks. A large amount of data is required in order to achieve a great level of generalization, which in turn will improve the overall classification accuracy on test data, that represent the actual data on which the trained model will make predictions [6]. However, obtaining this large amount of labeled data can be arduous, as the process of annotating large datasets is very time and resource consuming [2]. In some scenarios, it might even be impossible due to the lack of data and expert professionals to label the instances.

An alternative to labeling data is to use an already available, different but semantic-related, labeled dataset (Source domain) to train the network to make predictions on the target data (Target domain). A problem with this approach is that the traditional methods for training the neural networks rely on the i.i.d assumption, which states that the data used during training and the data on which the model will make predictions during test must be independent and identically distributed. Therefore, the shift in the data distribution between the source and target domains due to differences in image characteristics, such as illumination, quality and presence of background, will cause a considerable impact on the model's classification performance on the target data.

To overcome the difference between the source and target domains, an adaptation step is needed to diminish the effects of this domain shift and enable the model to have good results in the target data. This problem is known as Domain Adaptation (DA) and, as suggested in [7] and [2], it can be categorized based on the level of divergence between the domains and the availability of labeled data in the target domain. Concerning the divergence between the domains, the DA problem can be divided into: *Homogeneous*, in which the input space has the same dimensionality in both source and target domains, and the number of classes and each class concept do not change between the domains; and *Heterogeneous*, in which the input and label spaces can be different across the domains. The DA problem can be further categorized based on the availability of labeled data in the target domain: *Unsupervised*, where only unlabeled samples are available in the target domain; *Semi-supervised*, where, in addition to the unlabeled samples,

a small amount of labeled target samples are available; and *Supervised*, in which only a few target labeled samples are available and there are no target unlabeled data.

In this paper, we consider the image classification task and the Homogeneous and Unsupervised setting, in which the domains share the same label space, i.e. they have the same number of classes and each class definition is the same across the domains, and only unlabeled data are available in the target domain.

Many methods have been proposed in the literature to deal with the Homogeneous Unsupervised Domain Adaptation problem with visual domains [2], [8]. The strategies with the best performance have been using a domain discriminator in an adversarial framework during training to maximize domain confusion, thus minimizing the distribution shift [9], and pseudo-labeling heuristics that will automatically assign labels to target unlabeled data [10], [11]. In this work, we sought to enhance the performance achieved by the Robust Spherical Domain Adaptation (RSDA) method [10], a state-of-the-art method that uses both adversarial and pseudo-labeling strategies, by analyzing the effects of applying a dimensionality reduction algorithm to the feature space created by the network and changing how data from each domain is used during training.

The proposed modifications have been motivated by the hypothesis that, in a lower-dimensionality space, the shift between the domains would become less apparent as the *semantic-related* information would be privileged with lower dimensions instead of *image-conditions-related* information. Therefore, as in a homogeneous setting the shift between the domains is mainly caused by differences in image conditions, reducing the dimensionality would lead to better results. Futhermore, by using the data from both source and target domains in a more robust way, the whole adaptation procedure would become more effective.

Experiments in commonly used datasets in DA show that, in fact, our proposed modifications led to better results, with an increase of up to 7 percentage points in the target data classification accuracy when compared to the original method in some scenarios.

II. Related Work

In the recent years, many approaches have been proposed for solving the Domain Adaptation problem with visual domains and deep neural networks. As suggested by Wang and Deng [2], these approaches can be categorized between discrepancy-based, reconstruction-based and adversarial-based.

Discrepancy-based methods [12]–[14] incorporate a regularization term to the network, that is usually a distribution discrepancy metric, such as the Maximum Mean Discrepancy (MMD), to reduce the distribution gap between source and target samples and diminish the effects of domain shift.

Reconstruction-based methods assume that reconstructing target images, which is an unsupervised task, can lead to better adaptation. Some of these methods will use a multitask learning setup, adding a target image reconstruction task to the network, such as the Deep Reconstruction-Classification Networks (DRCN) [15] and the Deep Separation Networks (DSNs) [16], that are based on an Encoder-Decoder architecture.

Adversarial-based methods [9], [10], [17]–[19] use a domain discriminator in a two-player game to produce domain-invariant features. One of the main adversarial-based methods is the Domain Adversarial Neural Network (DANN) [9]. In DANN, a discriminator, similar to the one used in the Generative Adversarial Networks (GANs) [20], is added to the network pipeline. This discriminator receives as input the features produced by a feature extractor and outputs whether this feature comes from a source sample or a target one. A gradient reversal layer is added just before this discriminator, and on the forward pass it simply pass on the input received, while on the backward pass it multiplies the given gradient by a negative constant, thus directly implementing the min-max objective of the adversarial game between the discriminator and the feature extractor. Given this adversarial setup, the feature extractor will be encouraged to produce more domain-invariant features in order to maximize the discriminator loss, thus reducing the difference in the features' distributions across the source and target domains.

Some methods will combine the aforementioned approaches with some techniques such as the use of pseudo-labels [10], [11]. The idea behind this pseudo-label technique is to develop a heuristic that will automatically label the target samples. Then, these pseudo-labeled samples can be used during the adaptation procedure. The main difficulty lies on defining a strategy for assigning the pseudo-labels, as incorrectly labeled samples can severely impact the model's performance. Recently, Robust Spherical Domain Adaptation (RSDA) was proposed in [10]. This method is based on the aforementioned DANN [9] and uses a new robust pseudo-label loss formulation that better models the confidence of a given pseudo-label assignment being correct. In RSDA, the pseudo-labels are assigned using the output of a classifier trained on the usual DANN pipeline. Then, the probability of a pseudo-label being correctly assigned to a sample is estimated by Gaussian-uniform mixtures based on the cosine distance between the sample's feature representation and the centroid of each class. The mixtures' parameters are estimated using an Expectation-Maximization (EM) algorithm with the target data. Furthermore, RSDA performs the adaptation in spherical (L2-normalized) space, as the authors state it makes adaptation easier as difference in norms are eliminated. During training, after an initial stage, the network weights optimization and the pseudo-labels assignment and confidence estimation are executed alternately.

In this work we analyze the effects of changing how data from each domain is used in the previously discussed RSDA method and how would the incorporation of a dimensionality reduction strategy to the RSDA pipeline impact the performance on target data. Some methods in the literature also use dimensionality reduction to solve the adaptation problem

[21], [22]. In [21], the authors propose a two-stage feature-based adaptation approach based on optimal transport that employs dimensionality reduction with the goal to separate data samples as much as possible and enhance the feature discriminability of the source domain. In [22], the dimensionality reduction is used to map the produced features to a lower dimensional space in which the value of global and local metrics, such as the Maximum Mean Discrepancy (MMD), will be as small as possible, thus creating an ideal space for adaptation. The main difference between our work and the aforementioned ones is that we incorporate the pseudo-labeling strategy of RSDA [10], that proved to lead to great results, with the feature dimensionality reduction, based on the assumption, also made by the previous works, that a lower dimensional space would lead to better adaptation results.

III. Proposed Approach

In the homogeneous unsupervised domain adaptation setting, given a labeled source dataset $\{(x_i^s, y_i^s)\}_{i=1}^{N_s}$ and an unlabeled target dataset $\{x_j^t\}_{j=1}^{N_t}$, we want to train a convolutional neural network, comprised of a feature extractor F and a classifier C, that is able to correctly classify the target samples.

As noted in previous sections, the main goal is to diminish the effects of the domain shift caused by the difference in the data distributions between the source and the target domains. In this work, we propose modifications to the Robust Spherical Domain Adaptation (RSDA) [10] method with the goal to improve its results. In RSDA, a network comprised of a feature extractor F, a classifier C and a domain discriminator D, which will classify the samples based on their domain, is trained using a spherical adversarial training loss, as defined in Equation 1.

$$\mathcal{L} = \mathcal{L}_{bas}(F, C, D) + \mathcal{L}_{rob}(F, C, \phi) + \gamma \mathcal{L}_{ent}(F), \quad (1)$$

where the basic loss $\mathcal{L}_{bas}$ is based on the adversarial loss of DANN [9] and is defined as the sum of the cross entropy loss of source samples classification and the adversarial loss of the discriminator:

$$\mathcal{L}_{bas}(F, C, D) = \mathcal{L}_{src}(F, C) + \lambda \mathcal{L}_{adv}(F, D), \quad (2)$$

where λ is a negative constant. Notice that the adversarial goal is implemented directly by inverting the discriminator's classification loss. The authors of RSDA include a semantic matching loss (based on [23]) to this basic loss formulation. We, however, chose to not add this term in order to better analyze the effects of the proposed modifications to the original method.

The robust pseudo-label loss $\mathcal{L}_{rob}$ [10] is defined as

$$\mathcal{L}_{rob}(F, C, \phi) = \frac{1}{N_0} \sum_{j=1}^{N_t} w_\phi(x_j^t) \mathcal{J}(C(F(x_j^t)), \widetilde{y}_j^t), \quad (3)$$

where $N_0 = \sum_{j=1}^{N_t} w_\phi(x_j^t)$, $\mathcal{J}$ is the mean absolute error and $w_\phi(x_j^t)$ is

$$w_\phi(x_j^t) = \begin{cases} \gamma_j, & \text{if } \gamma_j \geq 0.5 \\ 0, & \text{otherwise} \end{cases}, \quad (4)$$

where $\gamma_j = P_\phi(z_j = 1 | x_j^t, \widetilde{y}_j^t)$ is the correct labeling probability associated to the sample x_j^t and the pseudo-label $\widetilde{y}_j^t$ that is estimated by the Gaussian-uniform mixture models using the parameters ϕ.

The training, as defined in [10], consists of an initial stage, where F, C and D are optimized using only the basic loss $\mathcal{L}_{bas}$. After that, the training procedure alternates between estimating the mixtures' parameters with the EM algorithm and optimizing the network with the complete loss $\mathcal{L}$, which includes the robust pseudo-label loss and an entropy-based loss $\mathcal{L}_{ent}$.

We change the way that the mixtures' parameters are estimated in the original work [10] by reducing the dimensionality of the features before calculating the distances relative to the centroids and by using data from both domains during this step, as in the original RSDA only data from the target domain are used during the EM algorithm.

The diagram presented in Figure 1 shows the order of the steps taken in the correct pseudo-labeling probability estimation procedure with the addition of the dimensionality reduction step. Futhermore, we can see that data from both source and target domains are used in the EM algorithm to estimate the optimal parameters of the mixtures. In the next subsections, we discuss these modifications and their motivations in detail.

A. Data Used During Mixture Parameters Estimation

In the original paper [10], the authors use K Gaussian-uniform mixture models, one for each class from a total of K classes, to predict the probability of a pseudo-label being correctly assigned to a sample. The pseudo-labels $\widetilde{y}_j^t$ are assigned to the target unlabeled samples x_j^t based on the output of the classifier C. Then, the probability of correct labeling P_ϕ is given by the respective mixture model

$$P_\phi(z_j = 1 | x_j^t, \widetilde{y}_j^t) = \frac{\pi_{\widetilde{y}_j^t} \mathcal{N}^+(d_j^t | 0, \sigma_{\widetilde{y}_j^t})}{\pi_{\widetilde{y}_j^t} \mathcal{N}^+(d_j^t | 0, \sigma_{\widetilde{y}_j^t}) + (1 - \pi_{\widetilde{y}_j^t}) \mathcal{U}(0, \delta_{\widetilde{y}_j^t})}, \quad (5)$$

where $d_j^t = \text{dist}(f_j^t, \mathcal{C}_{\widetilde{y}_j^t})$ is the cosine distance between the latent representation f_j^t of the j-th target sample and $\mathcal{C}_{\widetilde{y}_j^t}$ is the centroid of the assigned class $\widetilde{y}_j^t$. The parameters $\phi = \{\pi_k, \sigma_k, \delta_k\}_{k=1}^K$ of the K mixture models are estimated using an Expectation-Maximization (EM) algorithm. The main idea is that samples that are closer to the class centroid in the feature space are more likely to be correctly labeled and are modeled by the Gaussian component of the mixture. On the other hand, if the feature representation of a sample is far from the class centroid, then it is more likely to be incorrectly

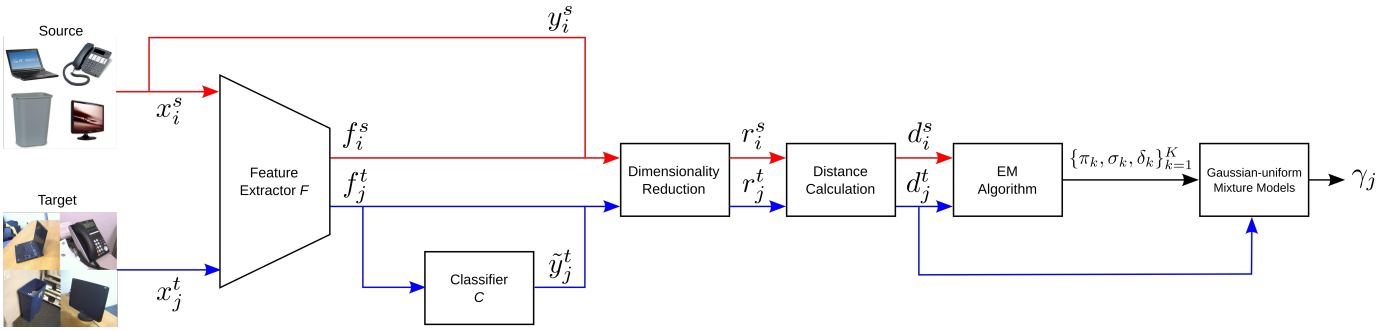


Fig. 1. Correct pseudo-labeling probability estimation procedure. First, the features from both source x_i^t and target x_j^t samples are extracted using the feature extractor F. Then, the pseudo-labels $\tilde{y}_j^t$ are assigned based on the classifier C output. A dimensionality reduction method is used to transform the features f_i^s and f_j^t into the lower-dimensional representations r_i^s and r_j^t. The ground-truth source labels y_i^s and target pseudo-labels may be used during the dimensionality reduction step, for example when we use the Partial Least Squares (PLS) algorithm. The cosine distance between each representation and the class centroids are then calculated. The Expectation-Maximization (EM) algorithm is executed on the calculated distances in order to obtain the optimal values of the parameters π_k, σ_k and δ_k for each mixture k. Finally, the Gaussian-uniform mixture models are used to estimate the correct pseudo-labeling probabilty γ_j of the target samples based on the distance to the class centroid calculated for each of these samples.

procedure TRAINING(F, C, D)
  Optimize F, C and D using the Basic Loss $\mathcal{L}_{bas}$ (Equation 2) for N_{epochs} epochs.
  for $stage = 1$ to N_{stages} **do**
    $\tilde{y}_j^t, \gamma_j =$ GETCORRECTPROBABILITY(F, C)
    Unfreeze and reinitialize the weights of F, C and D randomly.
    Train F, C and D with the $\mathcal{L}$ loss (Equation 1), for N_{epochs} epochs.

procedure GETCORRECTPROBABILITY(F, C)
  Freeze the weights of F and C.
  Obtain the features f_i^s and f_j^t produced by F for all Source x_i^s and Target x_j^t samples.
  Assign the pseudo-labels $\tilde{y}_j^t$ based on the output of C.
  Run a dimensionality reduction algorithm on f_i^s and f_j^t to obtain the reduced features r_i^s and r_j^t.
  Compute the centroid $\mathcal{C}_k$ for each class $k = \{1, 2, \ldots, K\}$ in the reduced space.
  Compute the distances d_i^s and d_j^t between each reduced feature and the respective class centroid (groundtruth labels for source samples and pseudo-labels for target ones).
  Run the EM algorithm, as defined in [10], to obtain parameters π_k, σ_k and δ_k for each k mixture model.
  Estimate the correct pseudo-labeling probability for the Target samples γ_j with the Gaussian-Uniform mixture models.
  return the estimated pseudo-labels $\tilde{y}_j^t$ with their respective probability γ_j.

Fig. 2. Algorithm detailing the proposed approach.

labeled and is modeled by the uniform component of the mixture.

In the original formulation of RSDA, only the features from target samples are used in the EM algorithm. This can lead to a loss of concept problem if a considerable amount of samples agree to an incorrect label, what would not be captured by the mixture models as the distance between the features and the centroids would be small. Therefore, the first proposed modification to the RSDA method consists of using data from both domains during the parameters estimation. As the ground-truth labels of the source samples are known, the estimation of the mixture parameters would be even more robust, thus making the deviation caused by incorrectly labeled target samples more unlikely.

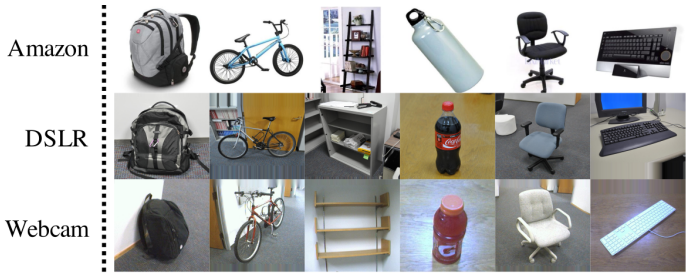
The use of data from both domains is possible due to the domain invariance promoted by the domain discriminator D, that is also used in RSDA as it is based on the DANN [9] method. As throughout the training D encourages the feature extractor F to produce domain invariant features due to adversarial game played between them, it is expected that the distribution of the produced features will not be too far apart, hence allowing the use of data from both domains in the EM algorithm.

In summary, we propose that, in contrast to what is done in the original RSDA method [10], both labeled data from source $\{(x_i^s, y_i^s)\}_{i=1}^{N_s}$ and pseudo-labeled data from target $\{(x_j^t, \tilde{y}_j^t)\}\}_{i=1}^{N_t}$ domains should be used during the mixtures parameters estimation with an EM algorithm in order to avoid class concept shift problems.

B. Dimensionality Reduction

We also propose to apply a dimensionality reduction method to the features produced by the feature extractor F before calculating the distances d that will be used during the mixtures parameters estimation. This is supported by the assumption, also made in [21], [22], that adaptation in lower dimensional spaces would be easier, as *semantic-related* information would be privileged with a lower amount of dimensions. Being that

(a) Office-Home dataset [24]. Image extracted from [24].



(b) Office-31 dataset [25]. Image adapted from [26].

Fig. 3. Datasets used in the experiments. Notice that even though the semantic meaning of each class (columns) is the same, the images conditions vary significantly across the domains (rows).

we consider the homogeneous scenario, in which the label space is the same across the domains and the domain shift is mainly caused by difference in image conditions, privileging information related to the semantic structure of the problem should lead to better adaptation results.

For our approach, given the samples x and their feature representation f, a dimensionality reduction algorithm is applied to f, generating reduced representations r. Then, the distance d input to the mixture models, as presented in Equation 5, is computed using the reduced features r.

C. Training

The proposed approach has the same training procedure as the original RSDA method [10], with the difference laying on how the mixtures' parameters are estimated. The steps are shown in Figure 2.

More details regarding the architecture used for each component and the values defined for the hyperparameters are given in the next section.

IV. Experimental Results

In this section, we evaluate the proposed modifications on two commonly used datasets in the Domain Adaptation literature, comparing the achieved results with the ones obtained by the original RSDA method.

A. Datasets

We evaluate the proposed approaches on the Office-31 [25] and Office-Home [24] datasets.

The Office-31 dataset contains 4,110 images of 31 categories of objects commonly found in an office environment. These images are distributed across 3 domains: Amazon, DSLR and Webcam. The Amazon domain consists of images scrapped from the online store, hence the pictures usually do not have a background and have more uniform illumination and quality. The DSLR and Webcam domains are comprised of images taken with a DSLR camera and a webcam, respectively, in an actual office. Therefore, the images from these two domains have a background and have more variation in quality and overall conditions, such as illumination and the pose of the object.

The Office-Home dataset is similar to the Office-31, but is bigger and more challenging. It has 15,500 images of 65 object categories divided in 4 domains: Artistic images, Clip Art, Product images and Real-World images. The main difference to the Office-31 dataset is the Clip Art domain, that consists of vector drawings of the objects, making it very challenging to adapt to real world images.

Some sample images from these datasets are presented in Figure 3. We follow the standard evaluation protocol for these datasets, in which the accuracy achieved on target data is reported by varying the source/target pairs.

B. Implementation details

The proposed modifications were implemented using PyTorch, based on the code distributed by the authors of RSDA [10] (Available at https://github.com/XJTU-XGU/RSDA). We use the ResNet-50 [3] with weights pre-trained on ImageNet as the feature extractor F. For the classifier C and domain

TABLE I
TARGET CLASSIFICATION ACCURACY (%) ON OFFICE-31 [25] DATASET. *EXTRACTED FROM [10]

	Amazon-DSLR	Amazon-Webcam	DSLR-Amazon	DSLR-Webcam	Webcam-Amazon	Webcam-DSLR
No Adapt (ResNet-50 [3])	80.00±3.56	79.66±0.29	59.46±2.41	91.40±1.07	61.88±0.90	99.00±0.82
DANN [9]*	79.7±0.4	82.0±0.4	68.2±0.4	96.9±0.2	67.4±0.5	99.1±0.1
RSDA [10]	90.63±0.09	92.03±0.74	72.25±1.11	97.61±0.18	75.74±0.55	**100.00±0.00**
RSDA + PCA	89.36±0.16	92.20±0.13	71.96±0.77	98.11±0.10	74.48±0.00	**100.00±0.00**
RSDA + PLS ($c = 10$)	90.03±0.34	93.75±0.21	71.00±0.33	97.90±0.16	75.01±0.42	**100.00±0.00**
RSDA + BOTH	93.04±0.34	93.42±0.33	75.79±0.46	98.66±0.47	77.52±0.83	**100.00±0.00**
RSDA + BOTH + PCA	92.30±0.25	92.96±0.88	75.93±0.39	**99.18±0.06**	77.92±0.12	**100.00±0.00**
RSDA + BOTH + PLS ($c = 10$)	**93.37±0.16**	**93.84±0.37**	**79.27±0.35**	99.16±0.12	**78.84±0.36**	**100.00±0.00**

TABLE II
TARGET CLASSIFICATION ACCURACY (%) ON OFFICE-HOME [24] DATASET. *EXTRACTED FROM [10]

	Art-Clipart	Clipart-Art	Clipart-Product	Product-Clipart	Product-Realworld
No Adapt (ResNet-50 [3])*	34.9	37.4	41.9	31.2	60.4
DANN [9]*	45.6	47.0	58.5	43.7	68.5
RSDA [10]*	51.50±0.05	**67.10±0.04**	72.10±0.02	51.10±0.05	81.08±0.06
RSDA + BOTH + PLS ($c = 10$)	**53.13±0.21**	65.68±0.19	**73.36±0.34**	**53.09±0.71**	**81.09±0.46**

discriminator D, we use the spherical layers and activation functions described in the original paper [10], with a bottleneck dimension of 256. All parameters, including the network learning rate and the EM-related ones are defined as in the original RSDA paper [10], in order to correctly evaluate the impact of the proposed modifications. We use the variation of RSDA based on the DANN [9] method. Therefore, we do not consider the semantic-matching loss.

C. Dimensionality Reduction

We evaluate the proposed addition of a dimensionality reduction step to the original RSDA pipeline with two popular algorithms: Principal Component Analysis (PCA) and Partial Least Squares (PLS). PCA reduces the dimensionality of data while preserving the data's variance by finding principal components, which are the dimensions with higher variance. PLS will build a lower dimensionality space by taking into account both the data and the labels. It will try to find the multidimensional direction in the data space which will better explain the variance in the label space.

As PLS takes into account the label information, we expect to see better results with PLS when compared to the ones obtained when using PCA. The 256-dimensional feature vector produced by the feature extractor F are reduced to $c = 10$ dimensions when using PLS. With PCA, the dimensionality is reduced until a threshold of 95% explained variance is met. We use the PCA and PLS implementations available in the scikit-learn package.

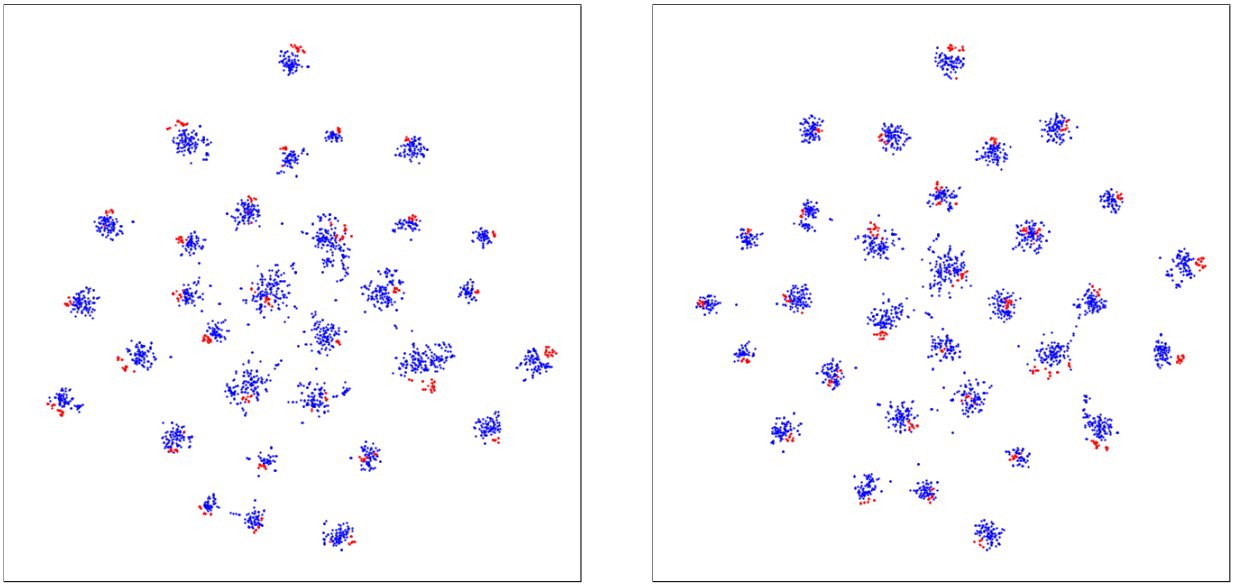
D. Results and Comparisons

The results achieved in the Office-31 dataset are reported in Table I. We compare the target classification accuracy achieved with each method. In the first and second rows, we show the results achieved when no adaptation is done, that is the model is trained using only source labeled data and no target data are used, and the result achieved with the DANN [9] method. Following, the results of the original RSDA method, our baseline, as reproduced by us using the implementation made available by the authors of [10], are presented.[1] Then, the results obtained when applying PCA or PLS in the original method are shown. Finally, in the last three rows of the table, we present the results obtained when using data from both source and target domains in the EM algorithm. The values displayed in the table are the average and standard deviation of the accuracy achieved on target data in three independent runs of each method.

According to the results in Table I, we can see that only the addition of the dimensionality reduction step to the original pipeline, using either PCA or PLS, did not result in a great improvement in the model performance on target data. However, when we combine the dimensionality reduction step and the use of data from both domains during the mixtures' parameters estimation, there was a significantly improvement in the model performance, specially in the *DSLR-Amazon* and *Webcam-Amazon* scenarios, that were the ones on which the original method achieved the worst results.

The results indicate that the proposed modifications, when combined, were able to improve the results achieved by the original RSDA [10] method, specially in the configurations where the discrepancy between the source and target domains are higher (DSLR-Amazon and DSLR-Webcam). This suggests that the use of source data in the Expectation-Maximization algorithm and the reduction of the feature dimensionality with PLS, a class-aware reduction, made the original method more resilient for situations in which the domains are very different.

[1]Even though we used the same architectures and parameters described in [10], we could not fully reproduce the results reported in the original RSDA paper. Here, we report the results we obtained with the parameters as described in the original RSDA paper.



(a) Original RSDA [10]. (b) Proposed RSDA + BOTH + PLS ($c = 10$).

Fig. 4. t-SNE [27] visualizations of the features produced by the feature extractor F after the complete training procedure on the DSLR-Amazon setting of the Office-31 dataset (Source - red, Target - blue). Notice how the proposed modifications to the original RSDA [10] method led to a better inter-class separability of the features, as the number of points in the low-density areas between the clusters is reduced.

Figure 4 presents the t-SNE [27] visualizations of the features produced by F after the complete training procedure in the DSLR-Amazon scenario. Notice how the proposed modifications led to features with a better inter-class separability in both source and target domains. This may be explained by the more robust way through which the correct labeling probabilities are estimated with the proposed approach.

The results obtained using the Office-Home dataset are presented in Table II. Due to resource and time restrictions, we report the results only on a subset of all possibles source-target pairs. Nonetheless, the selected pairs should illustrate how the proposed approach performs under different scenarios. In the first three rows of Table II we present the results with no adaptation, with DANN [9] and the original method. These results were extracted from [10]. In the last row of the table, we report the results achieved using the configuration that led to the best results in the Office-31 dataset.

Through the analysis of the results in the Office-Home dataset, we can see that the proposed modifications to RSDA also resulted in an improvement over the classification performance on target data achieved by the original method, albeit the improvement seen on this dataset was not as big as the one achieved in the Office-31 dataset. In the *Clipart-Art* configuration there was in fact a reduction in the accuracy on target data. Nevertheless, there was an improvement of up to 2% in the accuracy achieved in the other configurations.

The overall analysis of the results presented in Tables I and II shows that the proposed modifications led to an improvement over the original RSDA method, thus indicating that the hypothesis that guided this work do hold true. These results then demonstrate that it is beneficial for domain adaptation that we make an effective use of the available data in both source and target domains and that the transformation of the produced features to a lower dimensional space indeed leads to a better adaptation of the semantic knowledge across the domains.

V. Conclusion

In this paper, we presented two modifications to the state-of-the-art Robust Spherical Domain Adaptation (RSDA) [10] method with the goal to use data from the source and target domains in a more robust way and to analyze the impacts of adding a dimensionality reduction step to the original method's pipeline. Experiments conducted in commonly used datasets in domain adaptation showed that the proposed modifications indeed led to a better accuracy in target data in most of the scenarios, specially in those where the domains have very different conditions.

Aside from improving the results obtained by the original method in the selected datasets, this work also provided possible paths that could lead to an even more effective adaptation. Exploring further dimensionality reduction strategies and how to build the reduced space in order to better capture the semantic meaning that is shared by the domains could make adaptation even more robust. As the experiments' results show, there is still room for improvement in some scenarios in which no method can achieve a high accuracy on target data. Therefore, the search for more effective domain adaptation methods is still a very active research topic and the modifications described in this paper can certainly help guiding future works involving this subject.

ACKNOWLEDGMENTS

The authors would like to thank the National Council for Scientific and Technological Development – CNPq (Grants 438629/2018-3 and 309953/2019-7), the Minas Gerais Research Foundation – FAPEMIG (Grants APQ-00567-14 and PPM-00540-17). This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001. We also thank the support by the Coordination for the Improvement of Higher Education Personnel (CAPES) under Grant 88887.516264/2020-00 from the Public Security and Forensic Sciences (PROCAD) Notice.

REFERENCES

[1] K. Weiss, T. M. Khoshgoftaar, and D. Wang, "A survey of transfer learning," *Journal of Big Data*, vol. 3, no. 1, p. 9, May 2016. [Online]. Available: https://doi.org/10.1186/s40537-016-0043-6

[2] M. Wang and W. Deng, "Deep visual domain adaptation: A survey," *Neurocomputing*, vol. 312, pp. 135 – 153, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0925231218306684

[3] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[4] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[5] G. Huang, Z. Liu, G. Pleiss, L. Van Der Maaten, and K. Weinberger, "Convolutional networks with dense connectivity," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

[6] W. M. Kouw and M. Loog, "An introduction to domain adaptation and transfer learning," 2019.

[7] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[8] V. M. Patel, R. Gopalan, R. Li, and R. Chellappa, "Visual domain adaptation: A survey of recent advances," *IEEE Signal Processing Magazine*, vol. 32, no. 3, pp. 53–69, 2015.

[9] Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by backpropagation," in *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ser. ICML'15. JMLR.org, 2015, p. 1180–1189.

[10] X. Gu, J. Sun, and Z. Xu, "Spherical space domain adaptation with robust pseudo-label loss," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[11] G. Kang, L. Jiang, Y. Yang, and A. G. Hauptmann, "Contrastive adaptation network for unsupervised domain adaptation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4893–4902.

[12] M. Long, Y. Cao, J. Wang, and M. I. Jordan, "Learning transferable features with deep adaptation networks," in *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, 2015, pp. 97–105. [Online]. Available: http://jmlr.org/proceedings/papers/v37/long15.html

[13] M. Long, H. Zhu, J. Wang, and M. I. Jordan, "Unsupervised domain adaptation with residual transfer networks," in *Proceedings of the 30th International Conference on Neural Information Processing Systems*, ser. NIPS'16. Red Hook, NY, USA: Curran Associates Inc., 2016, p. 136–144.

[14] ——, "Deep transfer learning with joint adaptation networks," in *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ser. ICML'17. JMLR.org, 2017, p. 2208–2217.

[15] M. Ghifary, W. B. Kleijn, M. Zhang, D. Balduzzi, and W. Li, "Deep reconstruction-classification networks for unsupervised domain adaptation," in *Computer Vision – ECCV 2016*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds. Cham: Springer International Publishing, 2016, pp. 597–613.

[16] K. Bousmalis, G. Trigeorgis, N. Silberman, D. Krishnan, and D. Erhan, "Domain separation networks," in *Proceedings of the 30th International Conference on Neural Information Processing Systems*, ser. NIPS'16. Red Hook, NY, USA: Curran Associates Inc., 2016, p. 343–351.

[17] R. Li, Q. Jiao, W. Cao, H.-S. Wong, and S. Wu, "Model adaptation: Unsupervised domain adaptation without source data," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[18] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2962–2971.

[19] M. Long, Z. CAO, J. Wang, and M. I. Jordan, "Conditional adversarial domain adaptation," in *Advances in Neural Information Processing Systems*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., vol. 31. Curran Associates, Inc., 2018. [Online]. Available: https://proceedings.neurips.cc/paper/2018/file/ab88b15733f543179858600245108dd8-Paper.pdf

[20] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems*, Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger, Eds., vol. 27. Curran Associates, Inc., 2014. [Online]. Available: https://proceedings.neurips.cc/paper/2014/file/5ca3e9b122f61f8f06494c97b1afccf3-Paper.pdf

[21] P. Li, Z. Ni, X. Zhu, J. Song, and W. Wu, "Optimal transport with dimensionality reduction for domain adaptation," *Symmetry*, vol. 12, no. 12, 2020. [Online]. Available: https://www.mdpi.com/2073-8994/12/12/1994

[22] M. Jiang, W. Huang, Z. Huang, and G. G. Yen, "Integration of global and local metrics for domain adaptation learning via dimensionality reduction," *IEEE Transactions on Cybernetics*, vol. 47, no. 1, pp. 38–51, 2017.

[23] S. Xie, Z. Zheng, L. Chen, and C. Chen, "Learning semantic representations for unsupervised domain adaptation," in *International Conference on Machine Learning*, 2018, pp. 5419–5428.

[24] H. Demakethepalli Venkateswara, J. Eusebio, S. Chakraborty, and S. Panchanathan, "Deep hashing network for unsupervised domain adaptation," in *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, ser. Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017. Institute of Electrical and Electronics Engineers Inc., Nov. 2017, pp. 5385–5394.

[25] K. Saenko, B. Kulis, M. Fritz, and T. Darrell, "Adapting visual category models to new domains," in *Computer Vision – ECCV 2010*, K. Daniilidis, P. Maragos, and N. Paragios, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 213–226.

[26] X. Xu, X. Zhou, R. Venkatesan, G. Swaminathan, and O. Majumder, "d-sne: Domain adaptation using stochastic neighborhood embedding," in *The IEEE Conference on Computer Vision and Pattern Recognition*, June 16-20 2019, pp. 2497–2506.

[27] L. van der Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of Machine Learning Research*, vol. 9, no. 86, pp. 2579–2605, 2008. [Online]. Available: http://jmlr.org/papers/v9/vandermaaten08a.html